SciGRID_gas: The final IGGIELGNC-3 gas transmission network data set

Release 2.0

J.C. Diettrich,
A. Pluta, J.E. Sandoval, J. Dasenbrock, W. Medjroubi

Jul 01, 2021

CONTENTS

How to cite

J.C. Diettrich, A. Pluta, J.E. Sandoval, J. Dasenbrock, W. Medjroubi
SciGRID_gas: The final IGGIELGNC-3 gas transmission network data set
German Aerospace Center (DLR), Institute for Networked Energy Systems
Germany
doi: 10.5281/zenodo.4922529

Impressum

German Aerospace Center (DLR), Institute for Networked Energy Systems
Carl-von-Ossietzky-Str. 15
26129 Oldenburg
Germany
Tel.: +49 (441) 999 060

LIST OF FIGURES

LIST OF TABLES

Summary

This document describes the final resulting non-OSM data set "IGGIELGNC-3", where all missing values have been estimated using heuristic processes, and was generated by combining the following data sources:

- InternetDaten data set (**INET**) [DPM20e]
- Gas Infrastructure Europe data set (**GIE**) [GasIEurope20]
- Gas Storages Europe data set (**GSE**) [GasSEurope20]
- International Gas Union data set (**IGU**) [IGU20]
- EntsoG-Map data set (**EMAP**) [EntsoG20]
- Long-term Planning and Short-term Optimization data set (**LKD**) [KKS+17]
- Great Britain data set (**GB**) [nationalGrid20]
- Norway data set (**NO**) [Gassco20a]
- Consumers data set for Europe (**CONS**) [San21].

The goal of the SciGRID_gas project is twofold: a) to generate a comprehensive gas transmission network dataset for Europe and b) to develop and supply automated processes to create such data sets for Europe. Gas transmission networks and their data are essential for gas network modelling. The modelling community can derive major characteristics from such networks. Such simulations have a large scope of application, for example, they can be used to perform case scenarios, to model the gas consumption, to minimize leakages and to optimize overall gas distribution strategies. The focus of SciGRID_gas will be on the European transmission gas network, but the principal methods will also be applicable to other geographic regions.

Data required for gas transport models are the gas facilities, such as compressor stations, LNG terminals, pipelines etc. One needs to know their locations, in addition to a large range of attributes, such as pipeline diameter and capacity, compressor capacity, configuration etc. Most of this data is not freely available. However, throughout the SciGRID_gas project it was determined, that data can be grouped into two categories: a) OSM data, and b) non-OSM data. The OSM data consists of geo-referenced facility location data that is stored in the OpenStreetMap (OSM) data base, and is freely available. The OSM data set currently delivers highly accurate topological information on pipelines, however, does rarely contain any required meta information. The Non-OSM data set can fill some of those pipeline data gaps, and can additionally supply information such as pipeline diameter, compressor capacity and more. Part of the SciGRID_gas project is to mine and collate such data, and combine it with the OSM data set. Tools have been designed to fill data gaps and handle copy right issues. This will result in a complete gas network data set.

In this document, the chapter "Introduction" will supply some background information on the SciGRID_gas project, followed by the chapter "Data structure" that gives a detailed description of the data structure that is being used in the SciGRID_gas project. Chapter "Data sources" describes the different non-OSM input data sets: INET, GIE, GSE, IGU, EMAP, LKD, GB, NO and CONS. To estimate any missing data, the chapter "Heuristic methods" describes in detail, how missing attribute values (e.g. pipeline diameter) were generated. This is followed by the chapter "Final data set", which gives a brief overview on each set of components and in addition summarizes the changes to a previously published SciGRID_gas data set. In addition, a chapter has been added, in which the heuristic methods used to generate missing attribute values have been analysed in respect of their impact/sensitivity.

The data presented here, e.g. number of pipelines is valid for this current version, but might significantly change with future updates, as additional tools become available, or new relationships between data sets can be derived.

The appendix contains a glossary, references, graphical results of all heuristic attribute generation processes, location name alterations conventions and finishes with the table of country abbreviation.

CHAPTER

ONE

INTRODUCTION

SciGRID_gas is a three-year project funded by the German Federal Ministry for Economic Affairs and Energy [BMWi20] within the funding of the 6. Energieforschungsprogramm der Bundesregierung [BMWi11].

The goal of SciGRID_gas is to develop methods to generate and provide an open-source gas network data set and corresponding code. Gas transmission network data sets are fundamental for the simulations of the gas transmission within a network. Such simulations have a large scope of application, for example, they can be used to preform case scenarios, to model the gas consumption, to detect leaks and to optimize overall gas distribution strategies. The focus of SciGRID_gas will be the generation of a data set for the European Gas Transmission Network, but the principal methods will also be applicable to other geographic regions.

Both the resulting method code and the derived data will be published free of charge under appropriate open-source licenses in the course of the project. This transparent data policy shall also help new potential actors in gas transmission modelling, which currently do not possess reliable data of the European Gas Transmission Network. It is further planned to create an interface to SciGRID_power [MMK16] or heat transmission networks. Simulations on coupled networks are of major importance to the realization of the German *Energiewende*. They will help to understand mutual influences between energy networks, increase their general performance and minimize possible outages to name just a few applications.

This project was initiated, and is managed and conducted by the German Aerospace Center (DLR), Institute for Networked Energy Systems.

1.1 Project information

- **Project title**: Open Source Reference Model of European Gas Transport Networks for Scientific Studies on Sector Coupling (*Offenes Referenzmodell europäischer Gastransportnetze für wissenschaftliche Untersuchungen zur Sektorkopplung*)
- **Acronym**: SciGRID_gas (Scientific GRID gas)
- **Funding period**: January 2018 - July 2021
- **Funding agency**: Federal Ministry for Economic Affairs and Energy (*Bundesministerium für Wirtschaft und Energie*), Germany
- **Funding code**: Funding Code: 03ET4063
- **Project partner**: German Aerospace Center (DLR), Institute for Networked Energy Systems.

Gefördert durch:

aufgrund eines Beschlusses
des Deutschen Bundestages

1.2 Background

As of today, only limited data of the facilities of the European Gas Transmission Networks is publicly available, even for non-commercial research and related purposes. The lack of such data renders attempts to verify, compare and validate high resolution energy system models, if not impossible. The main reason for such sparse gas facility data is often the unwillingness of transmission system operators (TSOs) to release such commercially sensitive data. Regulations by EU and other lawmakers are forcing the TSOs to release some data. However, such data is sparse and too often not clearly understandable for non-commercial users, such as scientists.

Hence, details of the gas transmission network facilities and their properties are currently only integrated in in-house gas transmission models which are not publicly available. Thus, assumptions, simplifications and the degree of abstraction involved in such models are unknown and often undocumented. However, for scientific research those data sets and assumptions are needed, and consequently the learning curve in the construction of public available network models is rather low. In addition, the commercially sensitivity also hampers any (scientific) discussion on the underlying modelling approaches, procedures and simulation optimization results. At the same time, the outputs of energy system models take an important role in the decision-making process concerning future sustainable technologies and energy strategies. Recent examples of such strategies are the ones under debate and discussion for the Energiewende [BundesregierungDeutschland20] in Germany.

In this framework, the SciGRID_gas project initiated by the German Aerospace Center (DLR), Institute for Networked Energy Systems (Oldenburg, Germany) aims to build an open source model of the European Gas Transmission network. Releasing SciGRID_gas as open-source is an attempt to make reliable data on the gas transmission network available. Appropriate (open) licenses attached to gas transmission network data ensures that established models and their assumptions can be published, discussed and validated in a well-defined and self-consistent manner. In addition to the gas transmission network data, the Python software developed for building the model SciGRID_gas will be published under the GPLv3 license.

The main purpose of the SciGRID_gas project is to provide freely available and well-documented data on the European gas transmission network. Further, with the documentation and the Python code, users should be able to generate the data on their own computers.

The input data itself is based on data available from openstreetmap.org (OSM) under the Open Database License (ODbL) as well as Non-OSM data gathered from different sources, such as Wikipedia pages, fact sheets from TSOs or even newspaper articles.

The main workload of this project is to:

- retrieve the OSM and Non-OSM data sets for the gas infrastructure
- merge all available data sets

- build a gas transmission component data set
- generate missing data using heuristic methods
- document the process and the output.

The first step of the project was to collate a Non-OSM data set by searching the web for metadata that will be useful for the project. This included information, such as pipelines, compressors, LNG terminals, and their attributes, such as diameters, capacities etc. This data set is called the "InternetDaten" data set (INET). The raw data set has been published previously [DPM20e]. Additional data sets, such as the data from "Gas Infrastructure Europe" (GIE), "Gas Storages Europe" (GSE), "International Gas Union" (IGU) and the Norwegian gas transport system operator "Gassco" (NO) are also available. Here all those and other data sets have been merged. In addition, any missing values have been determined using heuristic processes. Other additional data sets will be merged at later stages, and will be made available through the project webpage.

This multi-stage release will allow us to easily and effectively incorporate feedback from potential users during the lifetime of the project. Those releases can be downloaded through the SciGRID_gas webpage with documentation, and can be seen as a snapshot of the current research project state.

Further information on the project can be found on the SciGRID_gas web page: https://www.gas.scigrid.de/pages/imprint.html.

The web page is maintained throughout the project lifetime, and will contain information on:

- General project information
- Contact details
- Presentations
- Bug/data fixes
- Data, code and documentation releases
- Publications.

As part of the SciGRID_gas webpage, one can also sign up to the SciGRID_gas newsletter by sending an email to

news.gas-subscribe@scigrid.de

1.3 Project goal

The overall goals of the SciGRID_gas project are:

- **Data output**: Creation of customisable gas transmission network data sets.
- **Open source**: Any one can download the data, make changes to it, pass it on to others, or even use it in commercial projects, as long as the SciGRID_gas project is mentioned as the original source of the data (CC by).
- **Application**: The outcome of the project can be used for a variety of scientific applications (e.g. sector coupling, entry-exit models etc.).
- **Transparency**: The Python code, the documentation and the data (that can be passed on under copyright licences) is supplied.
- **Extendibility**: Every user can extend the software code to their needs. However, we would encourage users to update and maintain the original git-repository and documentation for others.
- **Feedback**: Through constant data releases, it is hoped that the output data set will improve in quality and quantity by constantly incorporating feedback from the research community.

1.4 Document overview

This is an overview of this SciGRID_gas documentation, as this will help the user to better understand the overall project, its aims and the steps that were taken to obtain/model the resulting data set.

SciGRID_gas has been coded in Python, and hence, with that came the overall data structure that was selected for the project. As this is the most fundamental aspect for anyone wanting to use the data and the code, it is described first. In the chapter **Data Structure** we define the terms *Components*, *Elements*, and *Attributes*. We also give an overview on the internal workings of the SciGRID_gas source code.

A fundamental building block of the SciGRID_gas project is the data itself. Overall, we have classified the data into two groups: OSM and non-OSM data. The chapter **Data Sources** contains background information on the following data sets:

- InternetDaten data set (INET) [DPM20e]
- Gas Infrastructure Europe data set (GIE) [GasIEurope20]
- Gas Storages Europe data set (GSE) [GasSEurope20]
- International Gas Union data set (IGU) [IGU20]
- Norway data set (NO) [Gassco20a].
- Long-term Planning and Short-term Optimization data set (LKD) [KKS+17].
- Great Britain data set (GB) [nationalGrid20]
- EntsoG-Map data set (EMAP) [EntsoG20]
- Consumer data set (CONS) [San21].

Information is supplied on how the data was collated and how it was implemented into the SciGRID_gas data set structure. In addition, an overview of the extent of the data will also be given for the data, e.g. the number of elements or the list of attributes that the data contained.

The following chapter describes how the individual data sets were merged. Here elements of the component *LNGs*, *Compressors*, *Storages*, *Productions*, *PipeSegments* needed to be merged, if at least two of the raw input data sets contained some information on those components.

The single resulting data set contains a vast amount of data, however, a large part of the data was missing as well. Hence, the chapter **Heuristic Methods** will describe the different methods that have been implemented to estimate the missing attribute values.

The chapter **Final data set** contains a summary of the resulting data set, and attempts to give an overview of the results of the statistical methods applied from the previous chapter.

The chapter **Conclusion** briefly summarises the project and data set, which is followed by the chapter **Appendix** that contains sub-sections, such as *Glossary*, *References* etc.

1.5 Formatting style

Throughout this document certain editing format styles have been applied, to make it easier for the user to read the document.

Key SciGRID_gas component labels are written in italic, such as *PipeLines*, *Storages* etc.

Component attributes are also written in italic, such as *length_km*, *pressure_bar*.

Function names are written in bold, e.g. **M_CSV.read()**. This also includes build in statistical function, such as **mean** or **median**.

Directory names and file names are surrounded by double quotes, e.g. “StatsMethodsSettings.csv”.

CHAPTER

TWO

DATA STRUCTURE

A well designed and documented data structure is fundamental in any large-scale project. Good data structure in combination with tools, based on algorithms, improve the performance of any project output.

This structure needs to represent the gas flow facilities as good as possible. Hence, it needs to include components, such as pipelines, compressors etc. A finite number of components have been identified that are required as building blocks of a gas network. In addition, each component will contain attributes, such as pipeline diameter, maximal operating pressure, maximal capacity, number of turbines etc.

It is anticipated that the adopted data structure can be implemented in different types of gas flow models and will be used by the research community for topics, such as sector coupling or identifying gas transmission bottlenecks.

Within the SciGRID_gas project, the structure of the data model is part of classes defined within the Python code. Alterations may occur over the duration of the project, but it is envisaged that those will be small, and that compatibility will be assured.

The goal of this section is to describe in details the data structure that has been adopted and implemented into the Python code. This will be important in understanding other aspects of this document, such as exporting the data into CSV files or generating missing values.

2.1 Data structure description

This section contains information on the SciGRID_gas data structure, the format, and the code that can be used to import publicly available data into the project, so that it can be used in subsequent steps. Paramount for an understanding of the data structure is a good understanding of the terminology used throughout this section and the document in general. Hence, terminology will be introduced in the following sub-section.

2.1.1 Terminology

Throughout this document certain terms will be used, which will be described below and have been summarized in Figure 2.1.

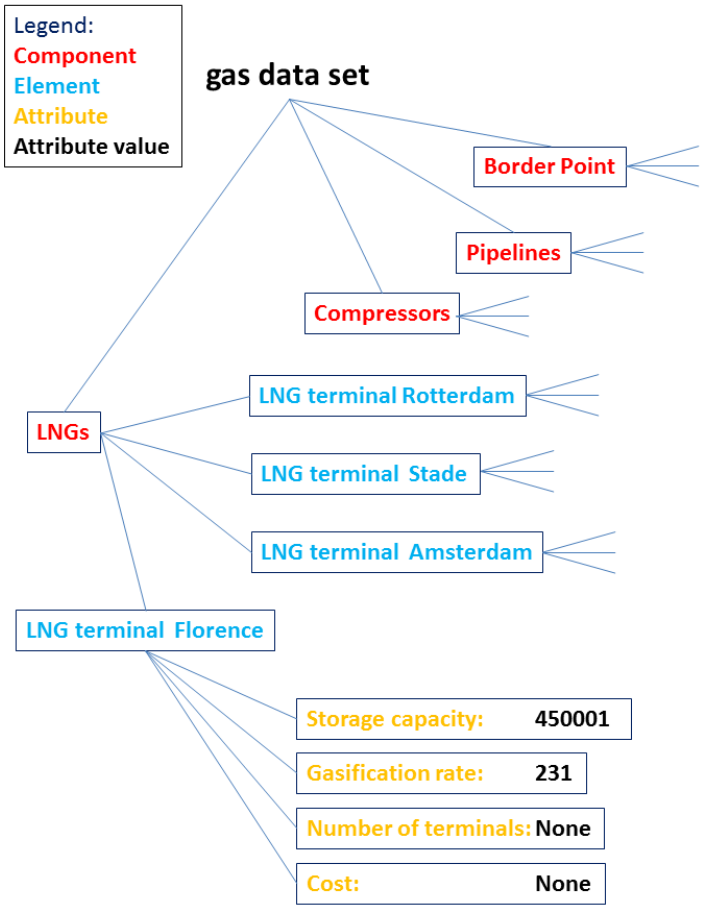


Figure 2.1: Data structure for the SciGRID_gas data set.

Gas transmission network

The term "gas transmission network" describes the physical gas transmission grid. This does not include the distribution of gas through gas distribution companies, but includes the long distance transmission of gas from producer countries to consumer countries, as carried out by the Transmission System Operators (TSO) [Wik20g]. In addition, throughout this document, the terms "transportation" and "transmission" are seen as interchangeable, and hence, will both be used describing the same.

Gas component data set

The term "gas component data set" is used for all raw data of objects/facilities that have been loaded using SciGRID_gas tools into a Python environment. Gas component data sets are used as input into our SciGRID_gas project. Several data sources can be loaded as gas component data sets, and then combined into a single gas component data set. However, not all elements (e.g. compressors) must be connected to pipelines. Hence, such a data set is referred to as a "gas component data set".

Gas network data set

A "gas component data set" can be converted into a "gas network data set", by connecting all non-pipeline elements to nodes and all nodes are connected to pipelines, and as part of the process all network islands have been connected or removed, resulting in a single network. Therefore, the network contains nodes and edges which are coherently connected, and all objects with the exception of pipelines are associated with nodes in this network, whereas pipelines are associated with edges.

Component

There are several component types in a gas transmission network, such as compressors, LNG terminals, or pipelines. In Figure 2.1 they are coloured red. Hence, whenever the word "component" is mentioned, it refers to one of these components. There are roughly a dozen different components that will form a gas network data set. They will be briefly explained below.

Element

The term "element" refers to individual facilities, e.g. the LNG terminal in Rotterdam, or the compressor in Radeland. In Figure 2.1 they are coloured blue. The first one is an element of the component LNG terminals, whereas the second one is an element of the component compressors. Hence, many elements make up a component. However, all elements are referring to different facilities by default. This means in a single network, one cannot have two elements of a component describing the same facility. The structure of elements is described below.

Attribute

"Attribute" is a term that is being used for the individual parameters that are associated with the elements. Examples of this term are gas "pipeline diameter", "maximum capacity", "max gas pipeline pressure", to name just a few and in Figure 2.1 they are coloured yellow. Overall there will be several hundred attributes in the SciGRID_gas project. However, the same attributes can occur in more than one component, e.g. "max flow capacity" exist for pipelines and also for compressors. Throughout the project, we have tried to keep the units of such attributes the same, so that there is no unit conversion required.

Attribute value

Each attribute has a value, most likely a number or a string. In Figure 2.1 they are coloured black. While booleans (*True*/*False*) are also allowed, more likely a "1" will stand for *True* and "0" for *False*. However, not all attribute values are given. Therefore, a no value for attribute values needs to be specified. In the SciGRID_gas Python code it is *None*.

The Figure 2.1 depicts the relationships between the terms "gas data set", "component", "element", "attribute", and "attribute value". As can be seen, a single gas data set consists of several components. On the next level, each component contains several elements. Further, each element has several attributes, where each attribute has a single or several values. The heuristic processes described in this document at a later stage will fill all missing values with heuristically generated values.

Gas component types

A gas transmission network consists of different components, such as pipelines, compressors etc. For the SciGRID_gas project a hand-full of components have been implemented, and will be described here briefly:

- *Nodes*: In a gas network, gas flows from one point to another point, which are given through their coordinates. All elements of all other components (such as compressor stations and power plants) have an associated node, which allows for the geo-referencing of each element. Overall the term *Nodes* will be used throughout this document, as it aligns with graph theory aspects.
- *PipeLines*: *PipeLines* allow for the transmission of the gas from one node to another. *PipeLines* are georeferenced by an ordered list of nodes.
- *PipeSegments*: *PipeSegments* are almost identical to *PipeLines*. However, are only allowed to connect two nodes. Hence, any *PipeLines* element (with 3 or more nodes) can easily be converted into multiple *PipeSegments* elements.
- *Compressors*: *Compressors* represent compressor stations, which increases the pressure of the gas, and hence, allows the gas to flow from one node to another node. A gas compressor station contains several gas compressors units (turbines).
- *LNGs*: *LNGs* is the acronym for the LNG terminals and LNG storages, which there are several in Europe, as some gas gets transported to Europe via ships.
- *Storages*: *Storages* are a further network component. Surplus gas can be stored underground (e.g. in old gas fields or salt caverns), and used during low supply or high demand periods.
- *Consumers*: *Consumers* is the term used for gas users, which can be households, industry and commercial. This data set will be generated through a Master project, and exludes power plants.
- *PowerPlants*: *PowerPlants* is the term used for gas use by power plants only.
- *Productions*: These can be wells inside a country where gas is pumped out of the ground. Most of the gas used in Europe comes from outside of the EU. However, there are several smaller gas production sites scattered throughout Europe.
- *BorderPoints*: *BorderPoints* are facilities at borders between countries, which are mainly used to meter the gas flow from one country to another.

Element structure

As described above, elements are describing individual facilities, such as compressors or LNG terminals. However, the overall structure of those elements is the same for all elements of all components, and is described as follows:

- *id*: A string that is the ID of the element, and must be unique.
- *name*: A string that is the name of the facility, such as "Compressor Radeland". In most cases this is not supplied.
- *source_id*: A list of strings that are the data sources of the element. As several elements from different sources could have been combined into a single element, one might need to know the original data sources.
- *node_id*: The ID of a geo-referenced node to which an element of the network is associated to. For a compressor, this will be just a single *node_id*. However, for a gas pipeline this entry would be a list of at least two *node_id* values: the starts node id and the end node id.
- *lat*: The latitude value of an element. For elements of type *PipeLines* and *PipeSegments*, *lat* is a list of latitude values. Throughout the SciGRID_gas project the projection World Geodetic system 1984 (epsg:4326) will be used.
- *long*: The longitude, analogue to lat.
- *country_code*: A string indicating the 2-digit ISO country code (Alpha-2 code, see Chapter 10.6 for list of countries and their codes) of the associated node of elements or list of nodes in case of *PipeLines* or *PipeSegments*.
- *comment*: An arbitrary comment that is associated with the element. In most cases this is not supplied.
- *tags*: This dictionary is reserved for OpenStreetMap data. It contains all associated key:value-pairs of an OpenStreetMap item.

In addition, there are three further groups of attributes to each element, which have been coded as Python "dictionaries", named:

- *param*
- *method*
- *uncertainty*.

The structure within each dictionary is the same. The dictionary *param* (short for "parameter") contains a list of attributes and their values. This list of attributes will be different for each component. For the component *PipeLines* they might be pipeline diameter, max pipeline pressure, and max pipeline capacity. For the component *Compressors* they might be, a number of turbines, overall turbine power, energy source of turbine or other.

The other two attribute dictionaries are *method* and *uncertainty*. Each of those two dictionaries contains exactly the same list of attributes as the *param* dictionary. However, their attribute values reflect the name of the dictionary. E.g. the attributes in the dictionary *method* contain the information on the method used to derive the attribute value that is stored in the *param* dictionary. Here methods of value generation can include heuristic methods names (in form of strings) that have been implemented in the SciGRID_gas project. However, if attribute values are not being generated by the SciGRID_gas project, but originate from one of the input data sources, then the attribute values in the *method* dictionary is set to "raw".

See example below, for an *LNGs* element with the following entries:

- "make_Attrib(const)": the attributes *end_year*, and *is_H_gas* have been set to a constant value
- "raw", indicating that the two attributes *max_cap_store2pipe_M_m3_per_d* and *start_year* contain original values
- "Lasso(max_cap_store2pipe_M_m3_per_d)", here for the attribute *median_cap_store2pipe_M_m3_per_d* a method was used that is based on the lasso method and uses the attribute *max_cap_store2pipe_M_m3_per_d* as input.

Similar is the content of the *uncertainty* dictionary. It contains information on the uncertainty of the attributes from the *param* dictionary of that component. Again, all attributes listed in the *param* dictionary are also present in the *uncertainty* dictionary. The attribute values here reflect the uncertainty of the attribute. Here, it is assumed that attributes with a method of "raw" have an uncertainty of zero. Only for those attributes, which were generated during heuristic SciGRID_gas methods an uncertainty larger than zero will be specified.

In addition, there are two special attributes in the param dictionary of the component *PipeSegments*: *path_lat* and *path_long*. These contain the waypoints between the start and the end node. They are separated in latitude and longitude, and are ordered in such a way, that they are following from the start node and go towards the end node.

2.2 Summary

The SciGRID_gas software is designed to construct a gas transmission network data set from different open and non-open source gas component data sets. The gas transmission data set needs to be available and stored in a precise and predefined way, which was described in this section. We have identified several *component*-types of a gas transmission network grid, like pipelines, compressor stations, LNG-terminals etc. Each specific facility that falls under such a component is considered an *element* of that component. Each element is described by a list of *attributes* and correspondent *attribute values*, including information on the uncertainty of the attribute value and the way the attribute value was generated.

CHAPTER

THREE

DATA SOURCES

Original data sets describing gas transmission networks are the property of the transmission system operators (TSOs) and are generally not freely available in the form and depth that is required for modelling purposes. The major reason for the difficulty of obtaining of such data is that most of the gas network infrastructure, namely pipelines, is buried underground. Thus, a pipeline diameter is hard to estimate locally. In addition, almost all of the data is commercially sensitive.

Nevertheless, some data is made available by gas transmission network operators, through different channels. E.g. information on the size and number of compressors could be made public through a press release, as part of a refurbishment. An example is given below (https://www.maz-online.de/Lokales/Teltow-Flaeming/Neue-Verdichterstation-entsteht-in-Radeland):

> "Die Eugal-Pipeline dient dazu, Gas aus der neuen Ostseepipeline Nord Stream 2 bis zur tschechischen Grenze zu leiten. 275 Kilometer von ihr verlaufen in Brandenburg. Grundsätzlich soll die neue Leitung parallel zur bestehenden Opal-Pipeline gebaut werden."

In addition, some information can be found on company web pages, (https://www.open-grid-europe.com/cps/rde/SID-752BB6B5-E0A975F2/oge-internet-preview/hs.xsl/NewsDetail.htm?rdeLocaleAttr=en&newsId=50190C3B-E14F-4685-9E64-E40EEAB57A28):

> "Open Grid Europe (OGE) is investing roughly EUR 150 million at its compressor station in Werne to improve the security and flexibility of energy supply for North Rhine-Westphalia and Germany. The upgrade of the station, which is one of the hubs of the pipeline network, will allow gas flows to be switched (reversed) from north to south and south to north. In addition, OGE is preparing the station for the upcoming transition from L- to H-gas. Through this fitness programme, the station's transmission capacity will increase by about 500,000 to 6.5 million m^3/h, which is equivalent to the annual consumption of more than 2,100 single-family homes. The project, which is due for completion at the end of 2018, is fully on track."

However, there is a public drive to gather such data and subsequently make it available. The major platform through which this is occurring is the Open Street Map database [Hel18]. OSM is a geo-referenced database through which people can supply geo-referenced information on all man-made and natural structures, ranging from mountains to buildings. To achieve this, people throughout the world wander the globe and geo-reference everything that they can find. This also includes gas-pipeline markers, compressor stations or LNG terminals. However, the major problem remains that one cannot measure or estimate the diameter of the underground pipelines, or the number and size of the compressor turbines, as compressors are within buildings, which are fenced off. Hence, such information is hardly supplied to the OSM platform.

For the reasons mentioned above, the available data can be separated into two different groups:

- OSM data: Data can be found in the OSM data base. OSM data is well geo-referenced, but contains little meta-information (information on the facility attributes, such as pipeline diameter or pipeline capacity). OSM data is very helpful to obtain accurate routes of pipelines.
- Non-OSM data: Non-OSM data have in general lower geographical accuracy but contain a lot of meta-information. Unfortunately, such information is only known for a few facilities. One exception to this rule

are shapefiles from TSOs. They are rare, but well geo-referenced. However, the resolution of the meta information can vary from TSO to TSO.

The following section will introduce non-OSM data sets, and at a later stage, this will be followed by a section on the OSM data.

3.1 Non-OSM data

Non-OSM data includes data from internet research, TSO press releases, TSO transparency platform, TSO public data, national open-source gas network data sets[1] etc.

Some of the TSO information had to be made available due to EU-regulations. Other information has been made public as part of a company's self-presentation and advertisement. The information used by the SciGRID_gas project focuses on:

- the quality of the data
- the format of the data
- the level of representation of the data
- and the copyright restrictions on the data.

In addition, each data source is unique. Source specific tools need to be developed, so that all data sources can be made accessible for the SciGRID_gas project.

A significant portion of the project was spent on finding non-OSM data sets. Further data sources might be available, but unknown to the authors. If the authors are made aware of additional sources, the project will try to incorporate those, as this would only increase the depth of the data available and increase the applicability of the gas network data set and model.

Non-OSM data sources are very specific, addressing only certain aspects of the entire gas infrastructure. E.g. the GIE [GasIEurope20] data set supplies information on the daily gas flow in and out of gas storages in LNG terminals. However, they fall short on specifying the fundamental information of the actual physical location. Other data sets, such as the LKD [KKS+17] data set is quite detailed in respect of pipelines, compressors and consumptions, however, only available for Germany.

Hence, the main task is to look closely at each data source, distil which data attribute values can be used, how it can be downloaded and incorporated into the SciGRID_gas model, and identify the copyright restrictions on the data source.

Due to copyright regulations, there are roughly two groups of data:

- Non-copyright restrictive data (N-CRRD): Here the copyright does not restrict the download, use and distribution of the data.
- Copyright restrictive data (CRRD): Here the data can be downloaded and used internally, but not re-distributed to others.

The following is a list of the data sources that will be used throughout the project and an identification into which group of copyright restriction they fall:

- **OSM** (https://www.openstreetmap.org) (N-CRRD)
- **GB** (https://www.nationalgridgas.com/land-and-assets/network-route-maps) (CRRD)
- **NO** (https://www.npd.no/en/about-us/information-services/available-data/map-services/) (N-CRRD)
- **LKD** (https://tu-dresden.de/bu/wirtschaft/ee2/forschung/projekte/lkd-eu) (N-CRRD)
- **ENTSOG** (https://transparency.entsog.eu/) (CRRD)

[1] An entire gas network data set is only available from the UK, see https://www.nationalgridgas.com/land-and-assets/network-route-maps'.

- **EMAP** (https://www.entsog.eu/sites/default/files/2020-01/ENTSOG_CAP_2019_A0_1189x841_FULL_401.pdf) (CRRD)
- **GIE** (https://www.gie.eu/) (N-CRRD)
- **GSE** (https://www.gie.eu/index.php/gie-publications/databases/storage-database) (N-CRRD)
- **IGU** (https://www.igu.org/) (CRRD)
- **INET** (see [DPM20e]) (N-CRRD)
- **CONS** (see [San21]) (N-CRRD).

Each data set and source comes with different copyright regulations. The copyright can be rather non-restrictive (e.g. INET) or can be restrictive (IGU). It is attempted to use only freely available data, so that such data can be re-distributed. In more restrictive data cases (IGU, GB), it is not allowed to download the data and distribute it to others. However, it is allowed to let other potential users know of the location of such data and supply them with tools that allow them to carry out the same data download and subsequent incorporation of the data into a gas network data set.

Note:

In case that other users are aware of other data sources that might be useful to this project, please get in touch and supply us with a brief description of the data and the location of such data, so that additional tools can be developed to incorporate the data in this project. Please use the following email address: developers.gas(at)scigrid.de

3.2 The InternetDaten (INET) data set

This section contains information on the generated content and nature of the so called **InternetDaten** data set (**INET**).

The INET data set is a special data set, as it was collated from many www sources and the information has been collated into CSV files. Please note that throughout the project, the separator within CSV files will need to be ";". This section here will give an overview on the INET data set, its components and how the data is stored in INET specific CSV files. Further the processing of the data in Python will be described.

Prior to the description of those processes, a general overview of the INET data set is given first, so that the reader has a better understanding of the size and depth of the data.

3.2.1 Overview of the INET data set

The INET data set contains geographical and meta information on gas facilities that were found through Internet searches. The data originated from www pages, such as Wikipedia, gas transmission system operators, fact sheets, press releases and more. Hence, most of the data had to be extracted manually out of text pages. To make this data available throughout the project, the data is being stored in CSV files. This also allows others to add additional properties and values to the INET data set at any stage. Tools have been written to load the INET from those CSV files and make them accessible throughout the project[1].

The Table 3.1 summarises the number of elements for each component that has been found so far. However, this does not imply that there is no missing data. In contrary, this data set comes with a lot of missing data.

Table 3.1: INET component summary.

Component Name	Count
BorderPoints	109
Compressors	248
PowerPlants	331
LNGs	33
Nodes	1397
PipeSegments	1053
Storages	199

In addition, a map (see Figure 3.1) visualizes these components for Europe.

3.2.2 Origin of the data

As has been stated before, the resulting INET data set originated from text sources found on the www. Here, for the pipeline JAGAL [Wik20h] an example from a Wikipedia page is given (https://en.wikipedia.org/wiki/JAGAL) in Figure 3.2:

As one can see, some information is given, such as location name of the compressor (Mallnow), total pipeline length (338 km), pipeline diameter (1200 mm) and maximum pipeline capacity (24 billion $\mathrm{m}^3\mathrm{a}^{-1}$). This is the information that is manually extracted from such pages and put into the CSV files.

To collate the data in an orderly manner, a system of CSV files has been created and will be described below.

[1] These tools will be made available during an upcoming release.

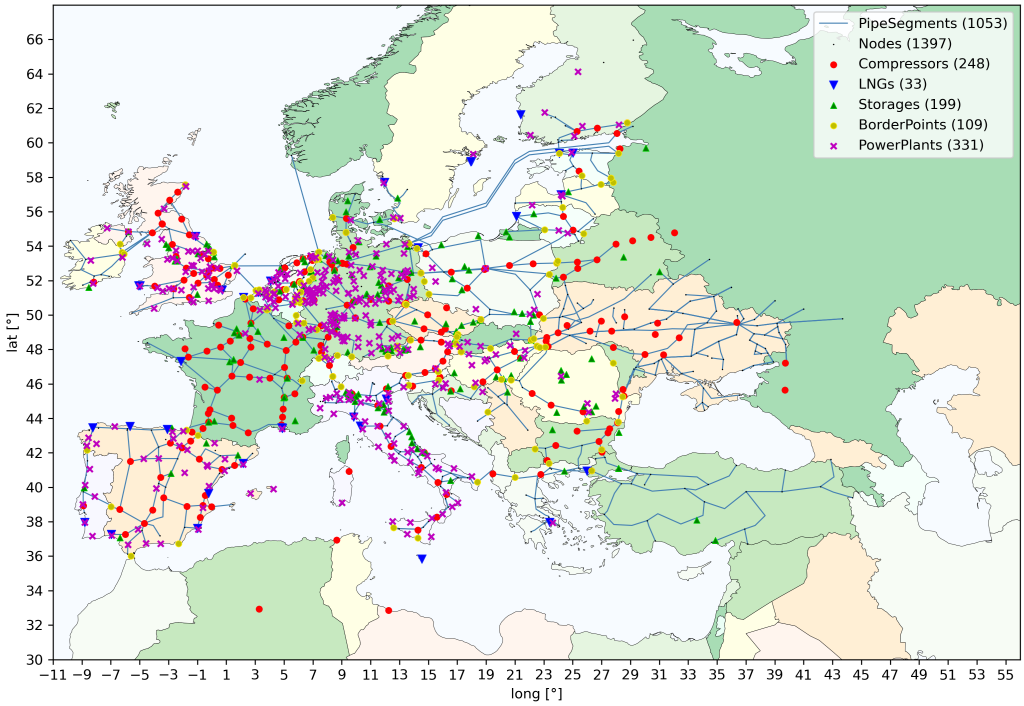


Figure 3.1: Map of the INET data set. The legend contains the number of elements for each component.

Commissioned	1999
Technical information	
Length	338 km (210 mi)
Maximum discharge	24 billion cubic meters per year
Diameter	1,200 mm (47 in)
No. of compressor stations	1
Compressor stations	Mallnow

Figure 3.2: Screenshot of part of the Wikipedia page for the pipeline JAGAL.

3.2.3 INET CSV file description

Each component of the INET data set is represented by a single CSV file. Each of those files has a single header line, and it is very important to know that entries in the first line should only be changed if one knows, what one is doing, as the first-row labels (the actual words) are imported and used as variable names in the SciGRID_gas project Python programs. Hence, if certain labels would be missing, the program would fail. In addition, each label needs to be unique within each file. It is advised to incorporate the units of the attributes into the label name, where possible.

Nodes.csv file

This is a unique file, and contains information on the nodes of the INET data set. Nodes are such entities, to which and from where pipelines can run, or to which other facilities can be associated to. Nodes supply information on a location including its name, its latitude and longitude, and the country in which it is located. Additionally, they supply information on the topological correctness of the lat/long values. The nodes component data is supplied to the SciGRID_gas data model only via a single CSV file, containing the following columns:

- *id*: A unique id of a node of type string. Most likely this will be the name of an element. White spaces are allowed in this string.
- *comment*: Here the user can place additional information on the location node.
- *country*: Here the user needs to write the 2 letter abbreviation of the country, in which this node is located (see Table 10.5 for a list of country codes used).
- *lat*: A number of the best estimate of the latitude of the location. Best latitude value (and long value) were attempted to be generated by using metadata of the facility node and satellite maps. Using the satellite data, address information etc., it was tried to visually find the facility of the node. Values need to be added as decimal degrees.
- *long*: This is analogue to *lat*.
- *node_id*: An identifier of a node.
- *source_id*: A unique identifier describing the source of the element. Here "INET" is the abbreviation for InternetDaten data set. Hence, all elements originating from the INET data set starts with the letters "INET".
- *name*: A string containing the name of the location. It is allowed to contain white spaces.
- *exact*: A number in the range of 1 to 5, indicating how accurate the lat/longs were supplied for the node. Options are as follow:
  - "1": The exact location of this node is known, as one was able to verify the facility through satellite data.
  - "2": Here the lat/long is not known exactly, however, one assumes that the location is within a small region (e.g. Krummhörn), hence, uncertainty not larger than 10 km.
  - "3": Here so little is known about the exact location, and one only knows that the location is within a large region (e.g. Hamburg). Hence, the actual location could be out by 10 km or more, but less than 100 km.
  - "4": Here so little is known about the exact location, and one only knows that the location is within a state (e.g. Niedersachsen). Hence, the actual location could be out by 100 km or more, but less than 1000 km.
  - "5": Here so little is known about the exact location, and one only knows that the location is within a country (e.g. Ukraine). Hence, the actual location could be out by 1000 km or more.

All other components need two files, the location file and the metadata file, which will be described next.

Compressor CSV meta file

The compressor file ("Meta_Compressors.CSV") contains all the metadata for the known compressor stations.

In addition to the seven mandatory columns introduced above, the following columns are currently implemented:

- *end_year*: Integer number of the year, when the operation of this element stopped. If it contains a value of "None", then it is still operational.
- *start_year*: Integer number of the year, when the operation of this element started. If it contains a value of "None", then this year is not known.
- *operator_name*: A string, containing the name of the operator of the compressor station.
- *pipe_name*: A string containing the label of the pipeline that the compressor is connected to.
- *source*: Information on where the information of this element originated from.
- *is_H_gas*: A boolean, indicating if the gas is of high calorific gas type ("1") or of low calorific gas ("0").
- *max_cap_M_m3_per_h*: A number, which is the overall capacity of gas that can be compressed by the compressor station. Values need to be supplied in units of $[\mathrm{Mm^3 h^{-1}}]$.
- *max_pressure_bar*: A number, which is the maximum pressure that the gas can be compressed to. Values need to be supplied in units of [bar].
- *max_power_MW*: A number, which is the sum of the power of all compressor units that are installed at the compressor station. Values need to be supplied in units of [MW].
- *num_turb*: The number of compressor turbines installed at the compressor facility. This number also includes the reserve turbine unit.
- *turbine_fuel_isGas_1*: A boolean, indicating if the turbine is powered by gas ("1"), or by electric ("0").
- *turbine_type_1*: A string containing additional information on the type of turbine unit, e.g. name of the turbine.
- *turbine_power_1_MW*: A number, indicating the power of the turbine unit. The value needs to be supplied in units of [MW].
- *turbine_fuel_isGas_2*: Information for the second turbine unit. Same as for *turbine_fuel_isGas_1* applies. Currently up to 6 individual units can be stored in the database, hence, the last digit in the identifier can be as large as 6.
- *turbine_type_2*: Information for the second turbine unit. Same as for *turbine_type_1* applies. Currently up to 6 individual units can be stored in the database, hence, the last digit in the identifier can be as large as 6.
- *turbine_power_2_MW*: Information for the second turbine unit. Same as for *turbine_power_1_MW* applies.
- ...
- *turbine_power_6_MW*: A number, indicating the power of the sixth turbine unit. The value needs to be supplied in units of [MW].

LNG CSV meta file

The LNG terminal metafile ("Meta_LNGs.CSV") contains all the metadata for the LNG terminals.

Next to the above described first seven columns the following columns are currently implemented:

- *end_year*: Integer number of the year, when the operation of this element stopped. If it contains a value of "None", then it is still operational.
- *start_year*: Integer number of the year, when the operation of this element started. If it contains a value of "None", then this year is not known. The values for this attribute dominantly originated from the King and Spalding report [KingSpalding18].
- *source*: Information on where the information of this element originated from.
- *max_workingGas_M_m3*: A number, indicating the maximum amount of liquid gas that can be stored, after having been brought in by ship. Values need to be supplied in units of [Mm^3] LNG. The values for this attribute dominantly originated from the King and Spalding report [KingSpalding18].
- *max_cap_store2pipe_M_m3_per_a*: A number, indicating the maximum amount of gas that can leave the LNG terminal. This gas is in gas phase. Values need to be supplied in units of [$\mathrm{Mm}^3\mathrm{a}^{-1}$]. The majority of those values originated from the EntsoB map [EntsoG20], and where missing were extracted from the King and Spalding report [KingSpalding18].
- *GCV_mean_kWh_per_m3*: This is the average Gross Calorific Value (GCV) of the gas flowing through this border point, retrieved from the Entso-G map [EntsoG20].
- *from_TSO*: This is a string of the name of the TSO that this LNG facility is associated to.
- *to_TSO*: This is a string of the name of the TSO that owns the pipe that the gas is fed into.
- *entsog_id*: A lot of the additional meta data originated from the data tables of the Entso-G map [EntsoG20]. Hence the EntsoG ID used on those tables has been place here, so that associating the meta data with the source is made easier.
- *max_vessel_size_LNG_m3*: Maximum size of ship that can land with size given in volume LNG.

BorderPoints CSV meta file

The metafile for *BorderPoints* elements ("Meta_BorderPoints.CSV") contains all the metadata for each border point.

Next to the above described first seven columns the following columns are currently implemented:

- *pipe_name*: A string, the name of the pipe that is passing the border point.
- *end_year*: Integer number of the year, when the operation of this element stopped. If it contains a value of "None", then it is still operational.
- *start_year*: Integer number of the year, when the operation of this element started. If it contains a value of "None", then this year is not known.
- *source*: Information on where some of the information of this element originated from.
- *entsog_id*: A lot of the additional meta data originated from the data tables of the Entso-G map [EntsoG20]. Hence the EntsoG ID used on those tables has been place here, so that associating the meta data with the source is made easier.
- *is_EU_to_EU*: Boolean (1 or 0) indicating if the border point is between two European Union nations or not.
- *from_country*: As the information content of the tables was flow of gas, a direction needed to be incorporated as well, to which the flow is related to. Here the from-country is given.
- *to_country*: Here the to-country is given, into which the gas is flowing and the gas flow numbers are related to.

- *from_TSO*: This is the name of the TSO from where the gas is being send from.
- *to_TSO*: This is the name of the receiving TSO.
- *max_cap_from_to_M_m3_per_d*: This is the maximum gas flow capacity in units of Mm^3/d that is flowing from the from-country to the to-country.
- *max_cap_to_from_M_m3_per_d*: This is the maximum gas flow capacity in units of Mm^3/d that is flowing from the to-country to the from-country. In case that it can be assumed that this border point allows only for uni-directional gas transmission, e.g. Nord Stream, then the value for *max_cap_to_from_M_m3_per_d* to zero. However, if the original data source was not clear enough on if there is gas flowing from the to-country to the from-country, then this field was left empty (None).
- *GCV_mean_kWh_per_m3*: This is the average Gross Calorific Value (GCV) of the gas flowing through this border point, retrieved from the Entso-G map [EntsoG20].

Storages CSV meta file

The metafile "Meta_Storages.CSV" contains all the metadata for gas storage elements within Europe.

Next to the above described first seven columns the following columns are currently implemented:

- *access_regime*: String indicating the access of the storage facility, TPA or not TPA (nTPA), and could be used for heuristic processes at a later stage.
- *end_year*: Integer number of the year, when the operation of this element stopped. If it contains a value of "None", then it is still operational.
- *start_year*: Integer number of the year, when the operation of this element started. If it contains a value of "None", then this year is not known. A value of 2050 was selected, if the site is in planing/construction, but not yet in operation.
- *store_type*: A string, indicating the type of storage, such as "Leeres Gas Feld" (empty gas field), "Salz Kaverne" (salt cavern) etc.
- *source*: Information on where the information of this element originated from.
- *is_H_gas*: A boolean that indicates if the gas is of high calorific nature ("1") or of low calorific nature ("0").
- *is_onShore*: A number, indicating if this gas store is on land or not. Options are "1": the gas store is on land; "0": the gas store is not on land, but off shore.
- *operator_name*: String, containing the name of the operator.
- *max_workingGas_M_m3*: A number indicating the maximum amount of gas that can be stored and worked with in that gas field. Values need to be supplied in units of $[\mathrm{Mm}^3]$.
- *max_cap_store2pipe_M_m3_per_d*: A number indicating the maximum amount of gas that can move from the gas store into a gas pipe. Values need to be supplied in units of $[\mathrm{Mm}^3\mathrm{d}^{-1}]$.
- *max_cap_pipe2store_M_m3_per_d*: A number indicating the maximum amount of gas that can move from the gas pipeline into a gas store. Values need to be supplied in units of $[\mathrm{Mm}^3\mathrm{d}^{-1}]$.

PowerPlants CSV meta file

The metafile "Meta_PowerPlants.CSV" contains all the metadata for gas power plants within Europe. The data used here is mainly a dump from the http://globalenergyobservatory.org.

Next to the above described first seven columns the following columns are currently implemented:

- *capacity_E_MW*: Value of installed electric power output in units of [MW].
- *capacity_TH_MW*: Value of installed thermal power output in units of [MW].
- *start_year*: Integer number of the year, when the operation of this element started. If it contains a value of "None", then this year is not known. A value of 2050 was selected, if the site is in planing/construction, but not yet in operation.
- *store_type*: A string, indicating the type of storage, such as "Leeres Gas Feld" (empty gas field), "Salz Kaverne" (salt cavern) etc.
- *source*: Information on where the information of this element originated from.
- *owner*: String, containing the name of the operator.

PipeSegments CSV meta file

The metafile "Meta_PipePoints.CSV" contains all the metadata for gas elements of type *PipeSegments* within Europe.

Next to the above described first seven columns the following columns are currently implemented:

- *is_bothDirection*: A boolean with value of '1' or '0'. If set to '1', then the gas pipeline can be operated in both directions, whereas if set to '0', then the gas can only flow from the start point to the end point. Hence, here the order of the *point-lables* in the pipes file is important.
- *length_km*: The overall length of the pipeline, and NOT of the segment. The value needs to be supplied in units of [km].
- *diameter_mm*: The diameter of the pipe in units of [mm].
- *max_pressure_bar*: The maximum pressure of the gas within the gas pipeline in units of [bar].
- *max_cap_M_m3_per_d*: The maximum annual gas volume that the pipe can transmit in units of $[\mathrm{Mm^3 d^{-1}}]$.
- *num_compressor*: The number of compressors along the pipeline.
- *end_year*: Integer number of the year, when the operation of this element stopped. If it contains a value of "None", then it is still operational.
- *is_H_gas*: A boolean that indicates if the gas is of high calorific nature ("1") or of low calorific nature ("0").
- *source*: Information on where the information of this element originated from.
- *lat_mean*: Average mean latitude value of the pipe-segment.
- *long_mean*: Average mean longitude value of the pipe-segment.

3.2.4 INET data density

"Data density" is defined as the ratio of the number of usable attribute values (not missing, e.g. filled or raw values) over the number of all elements of the component. Supposedly the INET would have two LNG terminals. One of the facilities has a known storage volume, whereas the other one does not. Hence, the data density would be 50% for the attribute storage volume. Here, the data density for the most relevant attributes will be given next for all components. At a later stage, missing values will be estimated through heuristic processes, to complete the data set.

PipeSegments elements

Overall, there are 1053 *PipeSegments* elements in the INET data set.

Table 3.2 summarizes the data densities for the most important pipe-segment attributes:

Table 3.2: INET *PipeSegments* data density

Attribute name	Data density [%]
diameter_mm	49
is_H_gas	95
is_bothDirection	8
length_km	100
max_cap_M_m3_per_d	13
max_pressure_bar	26
num_compressor	3

Compressors elements

Overall, there are 248 *Compressors* elements in the INET data set. The data densities for the most important attributes is given in Table 3.3 below:

Table 3.3: INET *Compressors* data density

Attribute name	Data density [%]
is_H_gas	99
max_cap_M_m3_per_d	7
max_power_MW	16
max_pressure_bar	7
num_turb	15
operator_name	11
pipe_name	10
turbine_power_1_MW	15
turbine_power_2_MW	14
turbine_power_3_MW	9
turbine_power_4_MW	3
turbine_power_5_MW	1
turbine_power_6_MW	0
turbine_fuel_isGas_1	14
turbine_fuel_isGas_2	14
turbine_fuel_isGas_3	9
turbine_fuel_isGas_4	3
turbine_fuel_isGas_5	1
turbine_fuel_isGas_6	0
turbine_type_1	9
turbine_type_2	9
turbine_type_3	6
turbine_type_4	3
turbine_type_5	1
turbine_type_6	0

Nodes elements

Overall, there are 1394 nodes. As described above, the information supplied is an "id", latitude and longitude values, the country code and a value indicating the accuracy of the node location. Hence, Table 3.4 summarizes the relative number of nodes within the possible value range of 1 to 5:

Table 3.4: Summary for the attribute *exact* of component *Nodes* of the INET data set.

Exact value	ratio of data with exact value [%]
1	60
2	27
3	5
4	3
5	4

Storages elements

Overall, there are 199 storage elements in the INET data set. The data densities for the most important attributes is given in Table 3.5 below:

Table 3.5: INET *Storages* data density

Attribute name	Data density [%]
access_regime	90
is_H_gas	17
is_onShore	96
max_cap_pipe2store_M_m3_per_d	70
max_cap_store2pipe_M_m3_per_d	71
max_workingGas_M_m3	81
operator_name	99
source	95
store_type	94

PowerPlants elements

Overall, there are 331 consumer elements in the INET data set. The data densities for the most important attributes are given in Table 3.6 below:

Table 3.6: INET *PowerPlants* data density

Attribute name	Data density [%]
PowerPlantType	34
est_generation_GWh	100
capacity_E_MW	100
capacity_TH_MW	100
is_gas_fuel1	100
is_gas_fuel2	12
is_gas_fuel3	0
is_gas_fuel4	0

A large portion of this data was derived from the Global Energy Observer web pabe (http:/globalenergyobservatory.org).

BorderPoints elements

Overall, there are 109 *BorderPoints* elements in the INET data set. The data densities for the most important attributes are given in Table 3.7 below:

Table 3.7: INET *BorderPoints* data density

Attribute name	Data density [%]
GCV_mean_kWh_per_m3	80
entsog_id	100
is_EU_to_EU	100
from_country	100
to_country	100
max_cap_from_to_M_m3_per_d	94
max_cap_to_from_M_m3_per_d	92
from_TSO	93
to_TSO	92

LNGs elements

Overall, there are 33 *LNGs* elements in the INET data set. The data densities for the most important attributes are given in Table 3.8 below:

Table 3.8: INET *LNGs* data density

Attribute name	Data density [%]
max_cap_store2pipe_M_m3_per_d	91
max_vessel_size_M_m3	67
max_workingGas_M_m3	94
GCV_mean_kWh_per_m3	64
entsog_id	67
from_TSO	67
to_TSO	67

Overall, a lot of data has been collated and is made available through the INET data set. However, as presented in the data density tables, a significant number of attributes have low data density. The following chapter in this document will demonstrate how missing values can be estimated, so that the generated SciGRID_gas data set has a data density of 100 % for each attribute.

3.2.5 Copyright and disclaimer for the INET data set

Copyright

the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit http://creativecommons. org/licenses/by/4.0/.

A list of the sources used for the generation of the INET data set can be found in Chapter 10.4.

Disclaimer

The INET data set is supplied on a best-effort basis only, using available information as documented gathered from the Internet. While every effort is made to make sure the information is accurate and up-to-date, we do not accept any liability for any direct, indirect, or consequential loss or damage of any nature–however caused–which may be sustained as a result of reliance upon such information.

3.2.6 Note on H-gas and L-gas

Almost all gas used in Europe is of high calorific nature (H-gas), with a gross calorific value (GCV) of around 11.367 kWhm^{-1}. However, due to a gas source exploited in the norther part of the Netherlands, some of the European gas consumed is of low calorific nature (L-gas) with a GCV of around 9.635 kWhm^{-1}. There are roughly 14.5 million customers in the Netherlands, Germany, Belgium and France that are being supplied with L-gas [Ent17]. As stated in [Ent17], the L-gas will be phased out by the end of 2030.

However, at the current state, there is still a significant portion of the gas network that is being used with the L-gas. Hence, information from [Ent17] and [Gas19] is being used to assign the correct GCV to the attribute *is_H_gas* of the component *PipeSegments*.

3.2.7 Improvements to previous release

Below major changes that occurred for version 2.0 :

- Component elements *Consumers* was changed to component *PowerPlants*.

Below major changes that occurred between version 1.0 and 1.1 are listed:

- The component *BorderPoints* was filled with additional information from the EntsoG map [EntsoG20].
- The component *LNGs* was filled with additional information from the Entso-G map [EntsoG20].
- Additional pipelines were added for the country of Ukraine.
- Additional pipelines were added for the country of Turkey.

3.3 Gas Infrastructure Europe (GIE) data set

Gas Infrastructure Europe (**GIE**) is a further dataset for the SciGRID_gas project which was generated by extracting information from the GIE web pages.

Gas Infrastructure Europe defines itself through the following statement:

> *'Gas Infrastructure Europe (GIE) is an association representing the sole interest of the infrastructure industry in the natural gas business, such as Transmission System Operators, Storage System Operators and LNG Terminal Operators. GIE has currently 68 members in 25 European countries.'* (https://www.gie.eu/).

Overall, GIE is the umbrella organisation for the following three gas components:

- **Storage**: GSE - Gas Storage Europe representing the Storage System Operators (SSO)
- **LNG**: GLE - Gas LNG Europe representing the LNG Terminal Operators (TO)
- **Transmission**: GTE - Gas Transmission Europe representing the Transmission System Operators (TSO).

The storage and the LNG information can be retrieved through an API supplied by GIE. However, there is no further information on the gas transmission part.

The APIs for the gas storage and the LNG terminals are:

- AGSI+ AGGREGATED GAS STORAGE INVENTORY (https://agsi.gie.eu/api/data/)
- Aggregated LNG Storage Inventory (ALSI) (https://alsi.gie.eu/api/data/).

Documentation for the APIs can be found on the web under: https://agsi.gie.eu/GIE_API_documentation_v001.pdf.

The GIE data set is copyright protected, hence, the SciGRID_gas project is not allowed to download the data for any other end user and pass it on to them. Hence, in the following subsections a method is being described that shows how to download the data and how to convert it into the SciGRID_gas data structure.

3.3.1 Requirements for accessing the GIE transparency platform

A private key is required for the GIE transparency platform so that one can download data from the GIE API.

As stated in the documentation for the GIE API:

> The API service is available to the public free of charge. Registration on the AGSI+ or ALSI website is mandatory for non-data providers to be able to use the API. Registration will result in a personal API key that is required within the API URL. The only purpose of this registration is to enable us to assess and improve the performance of our systems where and if required (user count, user activity, most popular data set types). Your account information and settings can be updated (and cancelled) at any time after signing in. Your data will be stored and securely handles as long as your account remains active.

For this you will need to go to the following link: https://agsi.gie.eu/#/login where on the right hand side you will need to fill in the registration details.

Under "Access to:" please select "Both AGSI+ and ALSI".

After registration you will have access to your private key. Copy the key and paste it into the following file:

/SciGRID_gas/Eingabe/GIE/GIE_PrivateKey.txt

This is your private key, hence, do not share it with others.

3.3.2 Data processing of the GIE data set

Gas infrastructure providers are requested to publish certain gas flow information. This data is accessible via the GIE URLs, and contains a vast amount of meta-data for storages and LNG terminals throughout Europe. Whenever the data is downloaded from the GIE API, the data needs modification, so that it is conform to the SciGRID_gas data model. Several tools have been written to achieve this. The GIE specific tools are described below for *Storages* and *LNGs*.

Processes for retrieving the data from the GIE API

First of all, one could access some meta-data on the storages and LNG terminals through the following internet links:

- LNG: https://www.gie.eu/index.php/gie-publications/databases/lng-database
- Storages: https://www.gie.eu/index.php/gie-publications/databases/storage-database

These data sets come as Excel sheets and contain information, such as name of facility, country, type of facility, and eic_code. Other information, such as max hourly capacity or LNG storage capacity, is also given, however, discarded due to copyright reasons.

LNGs elements

The following information are used from the LNG table above:

- country
- type
- eic_code
- short_name
- name
- nameShort.

This information (column heading and column data) needs to be placed into a CSV document. This file needs to be named "GIE_LNG.csv" and needs to be stored in the folder "/SciGRID_gas/Eingabe/GIE/".

Other additional fields from this table are:

- Region
- Status
- Investment
- Start-up year
- Type
- Operator short name
- Max. Hourly Cap. $[\mathrm{m^3(N)h^{-1}}]$
- Nom. Annual Cap. billion $[\mathrm{m^3(N)a^{-1}}]$
- Possible additional Nom. Annual Cap. billion $[\mathrm{m^3(N)a^{-1}}]$
- LNG storage capacity $[\mathrm{m^3LNG}]$
- Number of tanks
- Max. ship class size receivable $[\mathrm{m^3LNG}]$
- Number of jetties

- Min. sea depth alongside [m]
- Max. send out pressure [bar]
- TPA regime
- PCI list
- Operator long name.

This list can change during the lifetime of the project.

The EIC-code, facility code and country code is subsequently used to request time series information for each location from the GIE API.

The retrieved time series contain two useful values:

- the working LNG volume in the LNG storage tank
- the gas flow amount from the storage to the gas-pipeline, in units of GWh/d (see Table 3.9).

From the so created time series, one can determine the maximum working gas volume in the LNG storage tank in units of million LNG cubic meters. In addition, the maximum and medium gas flow from the storage to the gas pipeline is determined, in units of GWh per day.

Prior to estimating the maximum and median value from the retrieved time series, the time series was quality assured. This was done by removing any outliers/spikes.

Table 3.9: GIE incorporated attributes

Field identifier	Description	Units	Example
status	E (estimated) C (confirmed) N (no data)	E / C / N	C
gasDayStartedOn	The start of the gas day reported upon	YYYY-MMDD	2015-11-02
lngInventory	The aggregated amount of LNG in the LNG tanks at the end of the previous gas day	$1000\mathrm{m}^3$	5373.25
sendOut	The aggregated gas flow out of the LNG facility within the gas day	GWh/d	976.5
dtmi	Declared Total Maximum Inventory of LNG	$1000\mathrm{m}^3$	8898.99
dtrs	Declared Total Reference sendOut	GWh/d	6650.0
info	Service Announcement (if applicable)	URL	https://alsi.gie.eu/#/news/184

The following information is incorporated into the SciGRID_gas data structure: name, max storage volume, max and medium gas flow volumes, facility code, country code, and EIC-code.

Subsequently, the LNG storage volume and flow were converted to their corresponding gas phase values. The final units of the measurements were $[\mathrm{Mm}^3\mathrm{d}^{-1}]$. No geo-coordinates were given for LNG terminals within the GIE data set. This information has been retrieved from the INET data set by a comparison of the name and the country code of the facility.

Storages elements

A meta-data set for the storages is available as Excel book and can be downloaded from the following URL: https://www.gie.eu/maps_data/downloads/2018/Storage_DB_Dec2018.xlsx.

In this Excel book, the sheet "Storage DB" contains the following columns:

- Country: string indicating the country of the storage
- Concatenate: –missing description–
- Country Code: two letter acronym for country code
- Company code: number of company
- Facility code: code of the facility
- Operator: name of operator
- Facility/Location: name of facility location
- Status: status of storage unit (operational/under construction/planned)
- Investment: string indicating the investment (existing/expansion/new facility)
- Start-up year: year when started operation
- Type: storage type, e.g. depleted field, salt cavern,. . .
- Notes:
- onshore/offshore: either onshore of offshore location of the gas storage
- Working gas (technical) TWh: maximum working gas volume in units of [TWh]
- Working gas TPA TWh: maximum working gas volume under TPA in units of [TWh]
- Working gas no TPA TWh: maximum working gas volume not under TPA in units of [TWh]
- Withdrawal technical GWh/day: maximum withdrawal rate of gas in units of [GWh/d]
- Withdrawal TPA GWh/day: maximum withdrawal rate of gas under TPA in units of [GWh/d]
- Withdrawal no TPA GWh/day: maximum withdrawal rate of gas not under TPA in units of [GWh/d]
- Injection technical GWh/day: maximum injection rate of gas in units of [GWh/d]
- Injection TPA GWh/day: maximum injection rate of gas under TPA in units of [GWh/d]
- Injection no TPA GWh/day: maximum injection rate of gas not under TPA in units of [GWh/d]
- Access regime: access regime with two options "nTPA", "rTPA", and "No TPA"
- in EU28 number: string ("n" or "y") if part of the 28 EU members
- in EU28 SUM: string ("n" or "y") if part of the 28 EU members
- EU 28 filter: string ("NO" or "YES") if part of the 28 EU members.

A subset of the above data needs to be saved as column data into a CSV file, containing only the following parameters with their data:

- Country
- Type
- EIC-code
- Short Name

- Name
- nameShort.

This file needs to be named "GIE_Storages.csv" and needs to be stored in the folder "/SciGRID_gas/Eingabe/GIE/".

These is subsequently used to get access to the time series of the storage data set, by using the facility code, the country and the EIC-code.

All time series consist of the daily "working gas volume", the "daily injection capacity" and the "daily withdraw capacity". Maximum values for each of those parameters are extracted from those time series.

The same testing of the goodness of the data was carried out, as was carried out for the LNG data set.

In addition, gas flow values were converted from [$\mathrm{GWhd^{-1}}$] to [$\mathrm{Mm^3d^{-1}}$].

3.3.3 GIE data density

All GIE components (*Storages* and *LNGs*) have the following mandatory attributes:

- *id*: unique identifier
- *name*: name of the pipe-segment
- *source_id*: a source id
- *node_id*: the id of the start and the end node of the pipe-segment
- *lat*: a list of latitude values
- *longitude*: a list of longitude values
- *country_code*: a string pair indicating the country code of the start and the end point
- *comment*: a user comment.

LNGs elements

Overall, there are 21 *LNGs* terminals in the GIE data set. In addition to the default attributes, the following non-standard attributes (see Table 3.10) were supplied. The number of attribute values supplied for each attribute is given by the parameter "data density" (see Chapter 10.1):

Table 3.10: GIE *LNGs* data density

Attribute name	Description	Units	Data density [%]
eic_code	EIC code of LNG terminal		100
facility_code	unique facility code		100
max_cap_store2pipe_M_m3_per_d	maximum gas flow from storage to pipeline	$\mathrm{Mm^3d^{-1}}$	100
max_workingGas_M_m3	maximum stored gas in LNG terminals	$\mathrm{Mm^3}$	100
median_cap_store2pipe_M_m3_per_d	medium gas flow from storage to pipeline	$\mathrm{Mm^3d^{-1}}$	100
name_short	short name of the facility		100

Storages elements

Overall, there are 109 *Storages* facilities in the GIE data set. In addition to the default attributes, the following non-standard attributes (see Table 3.11) are supplied and populated with data:

Table 3.11: GIE *Storages* data density

Attribute name	Description	Units	Data density [%]
eic_code	EIC code of storage facility		100
facility_code	unique facility code		100
max_cap_pipe2store_M_m3_per_d	maximum gas flow from pipeline to storage	$\mathrm{Mm^3 d^{-1}}$	100
max_cap_store2pipe_M_m3_per_d	maximum gas flow from storage to pipeline	$\mathrm{Mm^3 d^{-1}}$	100
max_workingGas_M_m3	maximum working gas in storage	$\mathrm{Mm^3}$	100
name_short	short name of the facility		100

Nodes elements

Overall, there are 115 *Nodes* elements in the GIE data set. In addition to the default attributes, the following non-standard attributes (see Table 3.12) are supplied and populated with data:

Table 3.12: GIE *Nodes* data density

Attribute name	Description	Units	Data density [%]
exact	boolean indicating that storage is planned		100
eic_code	EIC code of storage facility		100
facility_code	unique facility code		100
name_short	short name of the facility		100
elevation_m	elevation at the node	m	100

Data availability and data usage

The API of the GIE web portal allows for the user to download time series on the daily gas amount (stored or available) for gas storages and LNG terminals. Here, we do not pass on the downloaded time series information, but only other data, which was derived from the time series information, such as maximum storage capacity.

3.3.4 Copyright

The generally valid copyright regulations for databases apply.

Data disclaimer

In addition, the data disclaimer is given as under: https://agsi.gie.eu/#/disclaimer:

> "All data is provided by the contributors on a voluntary basis and free of charge. The data provided by AGSI is for information purpose only. GSE is using reasonable efforts to invest in ensuring the correctness, completeness, and timeliness of the information provided herein. Data have been carefully checked, are updated at regular intervals and may be subject to changes, removal, or amendments without prior notice. GSE neither assumes any warranty or liability for the correctness and completeness of information/services and entries nor for the mode of presentation."

Acknowledgement

Here we would like to acknowledge GIE (Gas Infrastructure Europe with registered office at Avenue de Cortenbergh, 100 - B-1000 Brussels, Belgium).

3.3.5 Summary GIE data

The GIE data set supplies information on gas infrastructure facilities all over Europe, such as gas storages and gas LNG terminals. Data for those facilities are accessible by the SciGRID_gas software through special CSV data files that are downloaded from the GIE web page. The information in those facilities are automatically filtered and reshaped to the data structure of SciGRID_gas project. Units are partially converted to align with other project data. The facilities are further geo-reference by the SciGRID_gas software with the help of the INET data set.

Below a table summarises the number of elements for each component:

Table 3.13: GIE component summary

Component Name	Count
LNGs	21
Nodes	115
Storages	109

In addition, the map in Figure 3.3 visualizes the data for Europe.

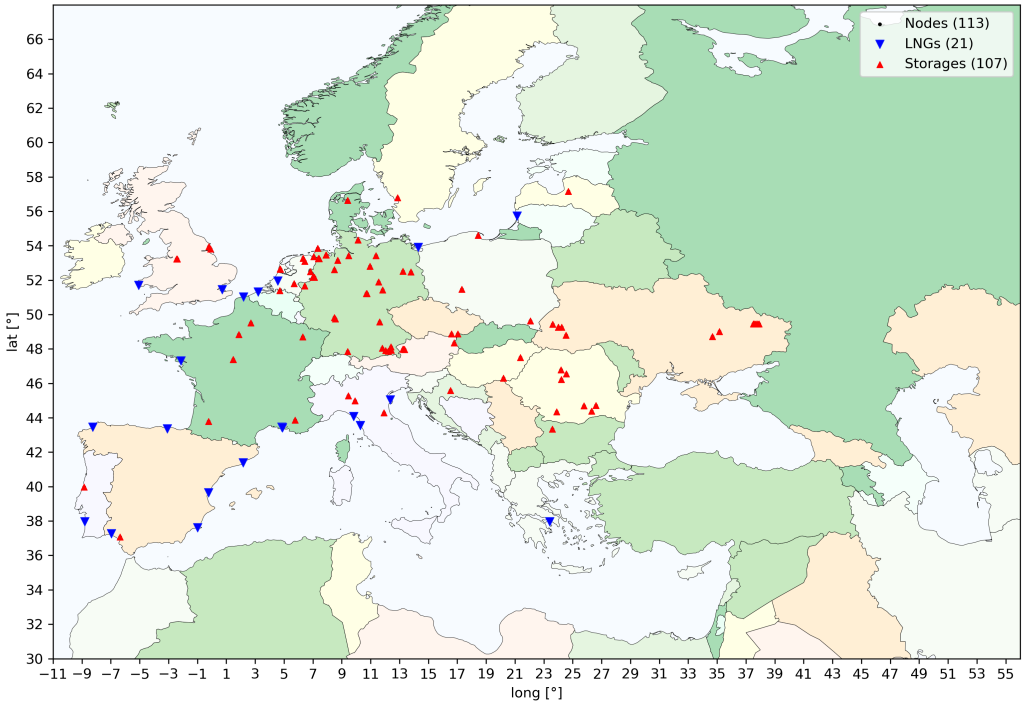


Figure 3.3: Overview map of the GIE data set for Europe.

3.4 The Gas Storage Europe (GSE) data set

This is the data set that was partially explained in the GIE section. However, the **Gas Storage Europe** (**GSE**) data set only contains information for the gas storage units, and contains slightly different information to the GIE data set. The GSE data set will be explained in this section here.

All together there were 254 storage facilities listed in the Excel book (see Chapter 3.3). This included planned and operational storage units, and storage units inside and outside of the 28 EU member states.

The Excel book can be downloaded from the following link:

https://www.gie.eu/maps_data/downloads/2018/Storage_DB_Dec2018.xlsx

which can be found on the following URL page:

https://www.gie.eu/index.php/gie-publications/databases/storage-database

3.4.1 Data processing of the GSE data set

Gas infrastructure providers are requested to publish certain gas flow information. This data is accessible through the GSE URLs, and contains a vast amount of meta-data for gas storages throughout Europe. However, whenever the data is downloaded from the GSE web page, the data needs modification, so that it conforms to the SciGRID_gas data model. Tools have been written to achieve this.

Overall, there is only one storage specific Excel book that could be downloaded. Table 3.14 contains a list of the columns from the Excel book, including the descriptions:

Table 3.14: Overview of GSE CSV data source

Field identifier	Description	Units	Used within SciGRID_gas
Country	string indicating the country of the storage		
Country Code	two letter acronym for country code		Y
Company code	number of company		
Facility code	code of the facility		
Operator	name of operator		Y
Facility/Location	name of facility location		Y
Status	status of storage unit (operational/under construction/planned)		Y
Investment	string indicating the investment (existing/expansion/new facility)		
Start-up year	year when started operation	yyyy	Y
Type	storage type, e.g. depleted field, salt cavern,...		
Notes			
onshore/offshore	either onshore of offshore location of the gas storage		
Working gas (technical)	maximum working gas volume in	TWh	
Working gas TPA TWh	maximum working gas volume under TPA	TWh	Y
Working gas no TPA TWh	maximum working gas volume not under TPA	TWh	
Withdrawal technical GWh/day	maximum withdrawal rate of gas	GWhd^{-1}	Y
Withdrawal TPA GWh/day	maximum withdrawal rate of gas under TPA	GWhd^{-1}	
Withdrawal no TPA GWh/day	maximum withdrawal rate of gas not under TPA	GWhd^{-1}	
Injection technical GWh/day	maximum injection rate of gas	GWhd^{-1}	
Injection TPA GWh/day	maximum injection rate of gas under TPA	GWhd^{-1}	Y
Injection no TPA GWh/day	maximum injection rate of gas not under TPA	GWhd^{-1}	
Access regime	access regime with two options nTPA, rTPA		
in EU28 number	char (n or y) if part of the 28 EU members		Y
in EU28 SUM	char (n or y) if part of the 28 EU members		
EU 28 filter	string (NO or YES) if part of the 28 EU members		
EU 28 filter	string (NO or YES) if part of the 28 EU members		

As was mentioned for the GIE data, no real lat/long values were given for the storage facility. Hence, the name matching with the INET data set (including the country code matching) was carried out. To achieve a better match, some of the names needed modification, such as substituting "HGas" and "H-Gas" with "H". In addition, parts of the location names were omitted, such as "SERENE Nord: ", "VGS SEDIANE B: ", "SERENE SUD" and "SEDIANE LITTORAL:".

Further, the gas flow and storage values supplied through the Excel book were in non-SciGRID_gas units, and the following gas properties were unit converted (see Chapter 10.2 for multiplication values used in the unit conversion process):

- '*max_cap_pipe2store_GWh_per_d*' to '*max_cap_pipe2store_M_m3_per_d*'
- '*max_cap_store2pipe_GWh_per_d*' to '*max_cap_store2pipe_M_m3_per_d*'
- '*max_workingGas_TWh*' to '*max_workingGas_M_m3*'.

3.4.2 GSE data density

The data of the GSE data set contains the following components:

- *Storages*.

The storage component will be described below.

As all components have the following attributes, they are presented here:

- *id*: unique identifier
- *name*: name of the pipe-segment
- *source_id*: a source id
- *node_id*: the id of the start and the end node of the pipe-segment
- *lat*: a list of latitude values
- *longitude*: a list of longitude values
- *country_code*: a string pair indicating the country code of the start and the end point
- *comment*: a user comment.

Storages elements

Overall, there are 210 usable *Storages* facilities in the GSE data set. In addition to the default attributes, the following non-standard attributes (see Table 3.15) are supplied and partially populated with data:

Table 3.15: GSE *Storages* data summary

Attribute name	Description	Units	Data density [%]
max_cap_pipe2store_M_m3_per_d	maximum gas flow from pipeline to storage	$\mathrm{Mm^3 d^{-1}}$	67
max_cap_store2pipe_M_m3_per_d	maximum gas flow from storage to pipeline	$\mathrm{Mm^3 d^{-1}}$	74
max_workingGas_M_m3	maximum working gas in storage	$\mathrm{Mm^3}$	78
name_short	short name of the facility		100
operator_name	name of the operator		100
start_year	year when the storage operation started	yyyy	87
status	indicating of storage is in operation, planned, or under construction		100

Nodes elements

Overall, there are 168 *Nodes* elements in the GSE data set associated with the 210 *Storages* facilities, meaning some storage facilities were associated to the same node element. In addition to the default attributes, the following non-standard attributes (see Table 3.16) were supplied and partially populated with data:

Table 3.16: GSE *Nodes* data summary

Attribute name	Description	Units	Data density [%]
exact	accuracy of node location		100
elevation_m	elevation at the node	m	100

3.4.3 Copyright and data disclaimer for the GSE data set

Data availability and data usage

The Excel book is available through the internet. However, no special copyright has been attached to the data set. Hence, normal copyright applies. Hence, we are not able to pass on the raw information that was downloaded by the SciGRID_gas project to others.

3.4.4 Copyright

The copyright regulations of this data can be found under (https://agsi.gie.eu/#/privacy-policy).

Data disclaimer

In addition, the data disclaimer is given as (https://agsi.gie.eu/#/disclaimer):

> "All data is provided by the contributors on a voluntary basis and free of charge. The Data provided by AGSI is for information only. GSE is using reasonable efforts to invest in ensuring the correctness, completeness, and timeliness of the information provided herein. Data have been carefully checked, are updated at regular intervals and may be subject to changes, removal, or amendments without prior notice. GSE neither assumes any warranty or liability for the correctness and completeness of information/services and entries nor for the mode of presentation."

3.4.5 Summary GSE data

The GSE data set summarizes information on gas storage facilities throughout Europe. Data for those facilities were accessible through an Excel book that was downloaded from the GIE web page. This data set applies to all of Europe, and special tools had to be written, to align their spatial data points to geo-reference location of the SciGRID_gas data set. Tools have been designed to convert the information from those CSV files and subsequently make them accessible for the SciGRID_gas project.

Table 3.17 lists the number of elements for each component found:

Table 3.17: GSE component summary

Component Name	Count
Nodes	168
Storages	210

In addition, the map in Figure 3.4 visualizes the data for Europe.

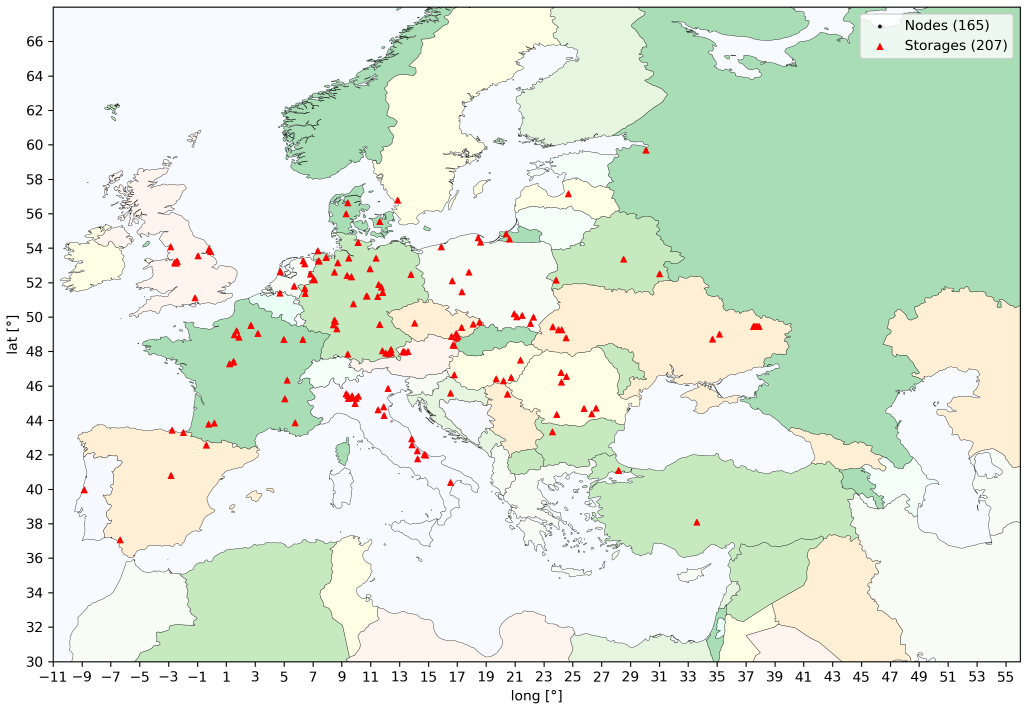


Figure 3.4: Overview map of the GSE data set for Europe.

3.5 The International Gas Union (IGU) data set

A further data set for storages stems from the **International Gas Union** (**IGU**). Their data are web based storage summary tables for roughly 170 European storage sites, which can be accessed through their online portal and link. The tables contain information on peak withdrawal capacity, injection capacity, and more. However, due to copyright limitations, the SciGRID_gas project is not allowed to pass on the actual downloaded values from those IGU tables. But this information will be used within the heuristic data generation processes.

3.5.1 Data processing of the IGU data set

IGU describes itself as follows (http://members.igu.org/old/about-igu):

> "IGU has more than 160 members worldwide and represents more than 97 % of the world's gas market. The members are national associations and corporations of the gas industry. The working organisation of IGU covers the complete value chain of the gas industry from upstream to downstream. As the global voice of gas, IGU seeks to improve the quality of life by advancing gas as a key contributor to a sustainable energy future. ... IGU seeks to collaborate with governmental agencies and multilateral organizations to demonstrate the economic, social and environmental benefits of gas in the global energy mix."

And its mission is to (http://members.igu.org/old/about-igu/vision-mission-and-objectives):

> "IGU is the key and credible advocate of political, technical and economic progress of the global gas industry, directly and through its members and in collaboration with other multilateral organizations. IGU works to improve the competitiveness of gas in the world energy markets by promoting transparency, public acceptance efforts and the removal of supply and market access barriers."

So the IGU data can be downloaded from their public internet port, not via an API, but through a normal HTML web page.

For the SciGRID_gas project tools were written that call those HTML web pages and then downloads the information that is required for the SciGRID_gas project. This was done for about 170 gas *Storages* sites throughout Europe.

Through the following URL http://members.igu.org/html/wgc2003/WGC_pdffiles/data/Europe/att one can access the HTML code for each individual storage unit. Hence, a tool was written that access those web pages and retrieves the data so that is fits into the SciGRID_gas data model project. Subsequently only information was used from those facilities that were not abandoned.

Here again, the lat/long values were not given for the individual storage locations, and in a further step, a lookup between the IGU and the INET data set was carried out, while considering the country code. 169 web pages were queried, resulting in 147 storage metadata datasets for Europe. This includes sites in Russia and other non-EU member countries. These non-EU storages were also downloaded, as it is envisaged that those non-EU datasets will help in the heuristic attribute generation process, and can be discarded at a later stage.

3.5.2 IGU data density

The data of the IGU data set contains only the component *Storages*, and its attributes will be described below.

Storages elements

Overall, there are 147 active *Storages* facilities in the IGU data set. 144 of those supplied usable information for the SciGRID_gas project. In addition to the default attributes, the following non-standard attributes (see Table 3.18) are supplied and partially populated with data:

Table 3.18: IGU *Storages* data summary

Attribute name	Description	Units	Data density [%]
max_cap_pipe2store_M_m3_per_d	Peak injection capacity, from pipe to storage	$\mathrm{Mm^3 d^{-1}}$	93
max_cap_store2pipe_M_m3_per_d	Peak withdrawal capacity, from storage to pipe	$\mathrm{Mm^3 d^{-1}}$	100
max_cushionGas_M_m3	Total cushion gas volume	$\mathrm{Mm^3}$	98
max_power_MW	max compressor power at storage facility	MW	75
max_storage_pressure_bphBar	Max allowable storage pressure	BHP bar	89
max_workingGas_M_m3	Installed max working gas volume	$\mathrm{Mm^3}$	100
min_storage_pressure_bphBar	Min storage pressure	BHP bar	71
net_thickness_m	Net thickness	m	53
num_storage_wells	No of storage wells/caverns associated with UGS		94
operator_name	Operator: String, of name of operator		100
permeability_mD	Permeability: floats, geological parameters, 10 - 1000	mD	56
porosity_perc	Porosity, floats, geological parameter, 15 - 22	%	53
start_year	Reference year: Integer of year	yyyy	100
storage_formation	Storage formation: sting describing geology of storage		59
store_type	Storage type: String indicating the storage type		100
structure_depth_m	Depth top structure (cavern roof)	m	95

Nodes elements

Overall, there are 137 node points in the IGU data set. In addition to the default attributes, the following non-standard attributes (see Table 3.19) are supplied and partially populated with data:

Table 3.19: IGU *Nodes* data summary

Attribute name	Description	Units	Data density [%]
exact	boolean indicating the accuracy of the lat/long values		100

3.5.3 Copyright and data disclaimer for the IGU data set

The copyright regulations of this data can be found under (http://members.igu.org/old/about-igu/legal).

In addition, the data disclaimer is given as under (http://members.igu.org/old/about-igu/legal).

3.5.4 Summary IGU data

The IGU data set supplies information on gas storage facilities. Data for those facilities are accessible through IGU HTML web pages that were accessed from the IGU public web pages. This data set covers continental Europe, and special tools had to be written, to align their spatial data points to geo-reference location of the SciGRID_gas data set. Tools have been designed to convert the information from HTML web page and subsequently make them accessible throughout the SciGRID_gas project.

Table 3.20 summarises the number of elements for each component found.

Table 3.20: IGU component summary

Component Name	Count
Nodes	137
Storages	144

In addition Figure 3.5 depicts a map of the IGU data for Europe.

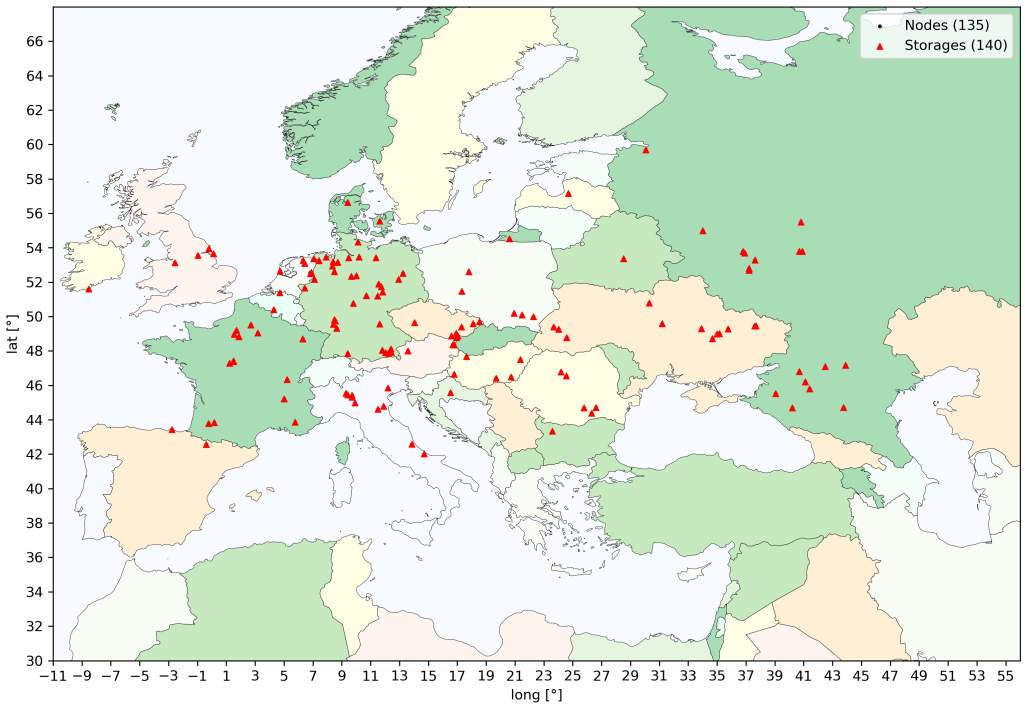


Figure 3.5: Overview map of the IGU data set for Europe.

3.6 EntsoG-Map (EMAP) data set

This section contains information on the content and nature of the so called **EntsoG-Map** (**EMAP**) data set, how this data was generated, its format, and its content.

3.6.1 Origin of the data

The origin of the EMAP data is a map in PDF format supplied by EntsoG. EntsoG is the acronym for "European Network of Transmission System Operators for Gas", and is an association of the European transmission system operators.

The EntsoG map covers all of Europe, including the non-EU states Russia, Ukraine, Belarus, Georgia, Azerbaijan, and others for the energy source gas. This map is being published on an irregular basis, and the latest version is from 2019. The project SciGRID_gas is very fortunate that a map version of the gas pipelines, drilling platforms and storage facilities is available. As part of the project, tools have been created to incorporate some of the information from the map into the project.

The latest map version of EntsoG is available from the following link: https://www.entsog.eu/sites/default/files/2020-01/ENTSOG_CAP_2019_A0_1189x841_FULL_401.pdf

The EntsoG map is freely available as a PDF file. Several steps need to be carried out to convert the PDF into the SciGRID_gas data structure. For this several Python tools have been created. However, this process cannot be fully automated. However, steps have been taken to automate as many aspects as possible, whereas some cleaning up will need to be carried out by the user by hand. The process of generating the data set is being described in more detail in Chapter 3.6.2, and information on the data density of the generated data set can be found in Chapter 3.6.3.

3.6.2 EMAP generation processes

Here a description is supplied, on how the data set was generated, originating from a PDF document and resulting in a SciGRID_gas data object. Lustenberger et al. [LSS+19] presented a similar pathway of dissecting the same data set, however, using the non-open tool ArcGIS. Here, the open source tool QGIS is being used.

Below is a general overview of the steps that have been implemented in converting the EntsoG PDF map into a single SciGRID_gas gas network data object:

- Separate the individual layers from the original PDF map into separate files (**PDF Layer generation**).
- Convert the above PDF files into high resolution TIFF files (**PDF to TIFF conversion**).
- Geo-reference the TIFF file, which resulted in raster layers (**Geo-reference of TIFF files**).
- Convert the raster layers into SciGRID_gas *PipeLines*, *Storages* and *Productions*, for all of Europe (**Generation of SciGRID_gas network elements**).
- Remove little pipelines that are assumed to be wrong artefacts of the PDF to TIFF conversion process (**Removing wrong elements**).
- Joining above data set into a single SciGRID_gas network data set, which will consist of many un-connected *PipeLines*, *Storages* and *Productions* (**Joining data**).
- Joining lose *PipeLines*, *Storages* and *Productions* to form one single SciGRID_gas network (**Generation of a single coherent SciGRID_gas data set**).

The overall outcome of this process is the conversion of the PDF map into more than 3000 *PipeSegments* elements, more than 200 *Storages* elements and more than 100 *Productions* elements throughout Europe, including Russia, and other non-EU states, resulting in a total length of more than 200000 km of pipes.

The steps of how to convert the PDF map into the SciGRID_gas data are presented here.

PDF Layer generation

The data source is a PDF map of the European gas transmission network, including sites of *Productions*, *Storages*, under-sea *PipeLines* and overland *PipeLines* in different thicknesses, based on their throughput. This map can be downloaded from the EntsoG web page (see link above). As the PDF document consisted of several layers, one can use an external tool to separate those layers and remove unwanted layers, such as legend, coastal lines, or gas fields. This process needs to be done by hand in an application, such as "Adobe Acrobat Reader". In this application, the layers tool can be selected, and individual layers can be saved as individual PDF layers. Below (see Figure 3.6) a screen shot shows the "Adobe Acrobat Reader" software with the EntsoG map loaded, and the layers tab expanded. Several layers can be seen in the screenshot and are part of the EntsoG map, such as "CAPDATA", "datapanel GRAY", ">>>LEGEND" etc.

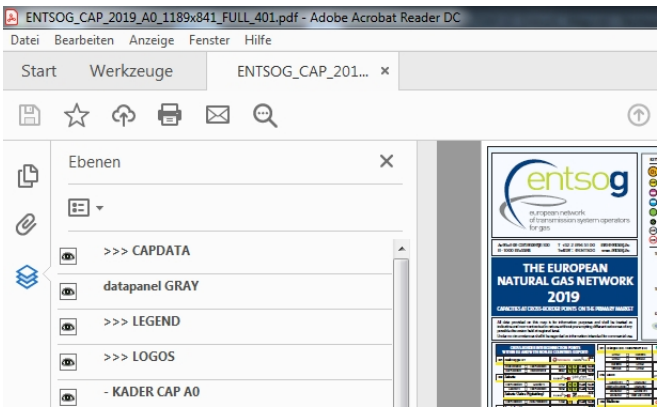


Figure 3.6: Screenshot of "Adobe Acrobat Reader" with the expanded layers tab, and a list of some layers to the left.

Most of the layers that are present in the EntsoG map do not contain information that is needed for this project, and can be discarded. But the following layers are required for the SciGRID_gas project:

- ">> STORAGE NONEU", will be part of the SciGRID_gas *Storages* component
- ">> STORAGE TYNDP", will be part of the SciGRID_gas *Storages* component
- "= DRILLPLATFORMS =", will be part of the SciGRID_gas *Productions* component
- "PIPELINES_NEW_GERMANY", will be part of the SciGRID_gas *PipeLines* component
- "PIPELINES > SMALL", will be part of the SciGRID_gas *PipeLines* component
- "PIPELINES > MEDIUM", will be part of the SciGRID_gas *PipeLines* component
- "PIPELINES > LARGE", will be part of the SciGRID_gas *PipeLines* component
- "=== NORTHSEA - pipes > GAS", will be part of the SciGRID_gas *PipeLines* component.

These layers needed to be exported individually into single PDF files.

In addition, further layers were needed for the geo-referencing process at a later stage:

- "BORDERS"
- "SHORES"
- "LANDMASS".

These three layers needed to be exported combined into a single PDF file, which will be referred to as the "*ENTSOG_Borders*" layer.

All resulting data files need to be stored in the following folder:

"../SciGRID_gas/Eingabe/Maps/EntsoG_2019/01_PDF/"

PDF to TIFF conversion

For geo-referencing of the Entso-G map information, the PDF files needed to be converted into TIFF format. For this an external application, such as https://onlineconvertfree.com/de/convert-format/pdf-to-tiff/, can be used. The user should select an application, which retains as much resolution as possible.

Resulting TIFF files need to be stored in the following folder:

"../SciGRID_gas/Eingabe/Maps/EntsoG_2019/02_TIFF/".

Geo-reference of TIFF files

As the projection of the original map is unknown, we need to determine the projection using an external application, such as QGIS. For this one needs to load the layer *ENTSOG_Borders* which was generated in a previous step above. The overall plan is to geo-reference this layer *ENTSOG_Borders* and in a second step apply the determined geo-referencing to all the actual gas facilities layers.

Hence, one needs to load the layer *ENTSOG_Borders* into QGIS, which is not geo-referenced at this stage. In addition, one needs to load a geo-referenced layer of Europe reference map or of the area of interest as well. Care needs to be taken that the reference map is projected in the projection that has been selected for the SciGRID_gas project. In the case for Europe, the projection "epsg:4326" was selected.

Now the following QGIS process is required: "Georeference GDAL". This is a plugin that can be installed from within QGIS, for QGIS versions of smaller than 3. For version 3.0 and newer, this plugin comes pre-installed with the base installation. (If you have problems finding "Georeference GDAL" in QGIS 3.x, then follow instructions under link https://gis.stackexchange.com/questions/274503/georeferencing-in-qgis-3-0).

The tool "Georeference GDAL" can be found under "Raster" and then "Georeferencer. . . ".

Here for SciGRID_gas the following steps need to be taken:

- Open QGIS
- Open a reference map of the European country layers, here the user can use the "TM_WORLD_BORDERS-0.3" layer [San19] that can be downloaded from the following site: https://koordinates.com/layer/7354-tm-world-borders-03/.
- Start the Georeferencer, and new Georeferencer window will open
- Press the [Open Raster] icon, and select the layer *ENTSOG_Borders*
- Open the "Transformation Settings" window by pressing the [Transformation Settings] icon, and select the following as depicted in Figure 3.7:

Here, the user needs to select the following:

- "Transformation type": "Thin Plate Spline"
- "Resemble method": "Cubic Spline"
- "Target SRS": "EPSG:4326 - WGS 84"
- "Output raster": "../SciGRID_gas/Eingabe/Maps/EntsoG_2019/03_Raster/ENTSOG_Borders.tiff"

and press the [OK] button to finish off this setup.

- In the "Coordinate Reference System Selector" select the "epsg:4326" coordinate reference system
- Select the [Add Point] icon
- By pressing the [Shift] button and using the mouse wheel, the user can find striking features on the TIFF map and select the location by pressing the left mouse button on top of it. Here as an example (see Figure 3.8) the

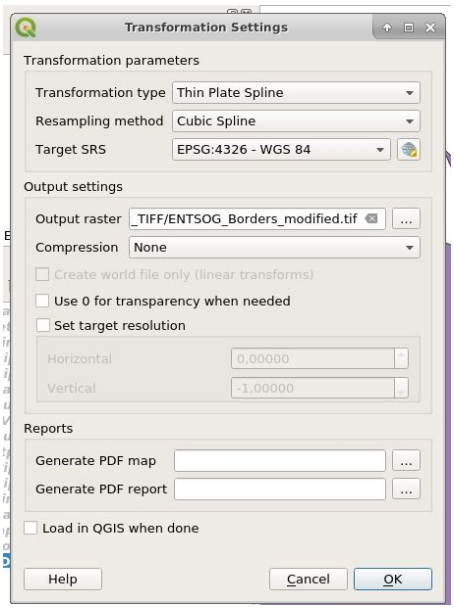


Figure 3.7: Screenshot of the "Transformation Settings" window.

border between Russia and Poland is being displayed, and a good location would be where the border meets the Baltic Sea.

Figure 3.8: Screenshot of a sample location of the Russian-Polish border in the Gdansk Bay.

- After pressing the left mouse button, the following window will appear (see Figure 3.9):
- Find on the loaded georeferenced map (e.g. TM_WORLD_BORDERS) the appropriate location and press the left mouse button again. This will populate the X/Y coordinates in the "Enter Map Coordinates" window, as shown in Figure 3.10.

Press the return button to lock in this geo-referenced pair of values.

- In the "Georeferencer" window an entry should appear in the "GCP table", where the table is located below the map (Figure 3.11).
- Repeat this process for a large number of points throughout Europe. Select points on the peripheries of Europe, but also select points within Europe, e.g. the three-border location of Belgium, Germany and the Netherland, or other territorial and geographical features, such as Isle of Guernsey or Isles of Scilly AONB. However, try not to use too many point pairs, a good spread is more important. (Here about 200 points were selected in the original

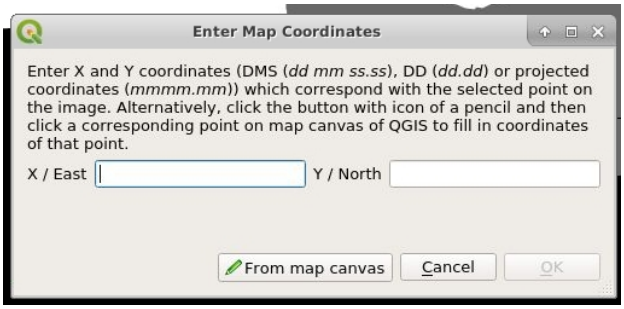


Figure 3.9: Screenshot of the new window "Enter Map Coordinates".

Here the user needs to press the [From map canvas] button.

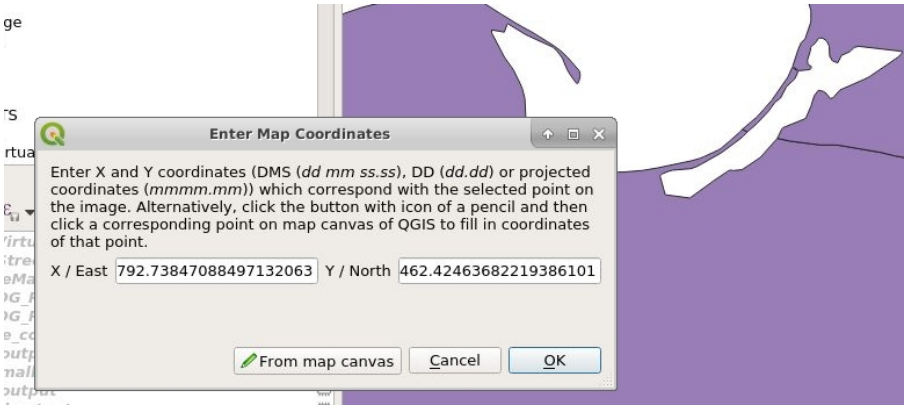


Figure 3.10: Screenshot of the window "Enter Map Coordinates" with the populated X/Y values.

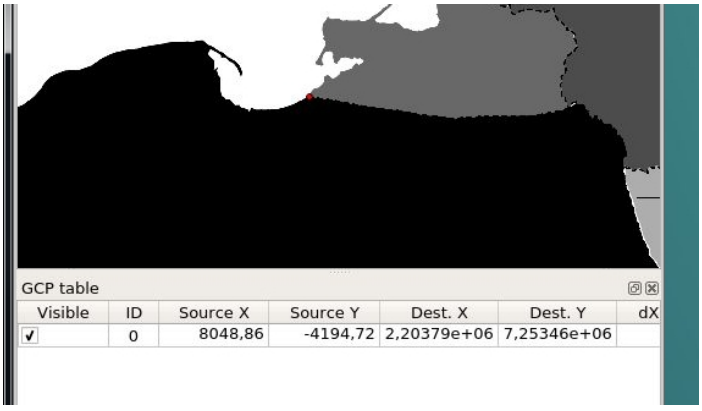


Figure 3.11: Screenshot of the "GCP table" entry with the new pair of coordinates, within the "Enter Map Coordinates" window.

process.)

- Now the user can check the geo-referencing by pressing the [Start Georeferencing] icon. This process might take several minutes. It will result in a new layer in the QGIS Layers list. Try to visualize this new layer and the reference map (e.g. "TM_WORLD_BORDERS-0.3"), by setting the top layer slightly transparent, so that one can eye up the newly projected layer "ENTSOG_Borders" with the reference map layer and look for areas of large difference (example given in Figure 3.12). Now more pair points can be added to rectify areas of imperfect geo-transformation, until the user is satisfied with the result. As an example, Luxembourg is presented here, and one can see that the locations of the border triangle of Luxembourg, the Netherlands and Germany on the north and Luxembourg, the Netherlands and Belgium on the west are not perfect. Hence, placing additional geo-referencing pairs might help to rectify this discrepancy.

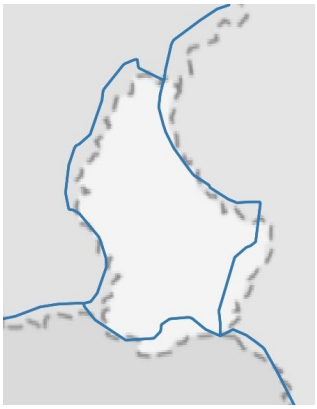


Figure 3.12: Screenshot of both layers around Luxembourg, showing the mismatch of the transformation.

- If the user is satisfied with the geo-referencing and the underlying pairs of values, the user needs to save the point pairs, as they will be used for the other TIFF layers. This can be achieved by pressing the [Save GCP Points as] icon in the "Georeferencer - . . . " window. A window will pop up and the user will need to enter a location and a file name of the points pair table.

Now the user needs to apply this same geo-referencing to the other layers of the EntsoG map. For this carry out the following steps:

- Select a new TIFF file from the layer list above in the "Georeferencer - . . . ".
- Open the previously saved GCP table by pressing the [Load GCP Points] icon.
- Select a different destination file under the [Transformation Settings] window.
- Initiate the geo-referencing process by pressing the [Start Georeferencing] icon.

This should be carried out for the *PipeLines*, *Storages* and *Productions* layers.

The overall output will be that the user will have created raster TIFF layers of the EntsoG gas elements, such as *PipeLines*, *Storages* and *Productions*, which are geo-referenced. Those resulting files should be stored in the folder: "../SciGRID_gas/Eingabe/Maps/EntsoG_2019/03_Raster/".

Generation of SciGRID_gas network elements

In the next step, the user needs to convert the raster layer into SciGRID_gas elements. To achieve this, Python code can be executed by the user.

The Python routines are combined in the **M_Maps** module, and can be accessed with the **M_Maps.read()** function. A large list of settings is required. However, they have been implemented into the code as default values, if no other values are supplied.

The previous steps created a geo-reference raster layer. However, this raster layer needs to be converted into polygons, which subsequently need to be converted into SciGRID_gas *PipeLines*, *Storages* and *Productions* elements.

The functions that have been developed to carry out those transformations are listed below for each sub-section.

Raster to polygons

The main function that is being used to convert the raster files into polygons is called **M_Maps.raster2Polygon()**. This function uses the freely available **GDAL** Python module that can be downloaded and installed for Python. The resulting file format is of type shapefile, and resulting files will be stored in the folder "../SciGRID_gas/Eingabe/Maps/EntsoG_2019/04_Polygon/".

The above process created a very large number of polygons, where some of the polygons are of the size of the PDF raster scanning resolution. To reduce the number of polygons, horizontally adjacent polygons are combined into single polygons, reducing the number of polygons by about 25 %. Results of this process are written as shapefiles into the folder "../SciGRID_gas/Eingabe/Maps/EntsoG_2019/05_Polygon/".

Manual shapefile clean-up processes

After the above step the user needs to carry out a manual process. This is required, as the polygons created contain spatial "mistakes", as polylines are surrounded by polygons, as can be seen in Figure 3.13.

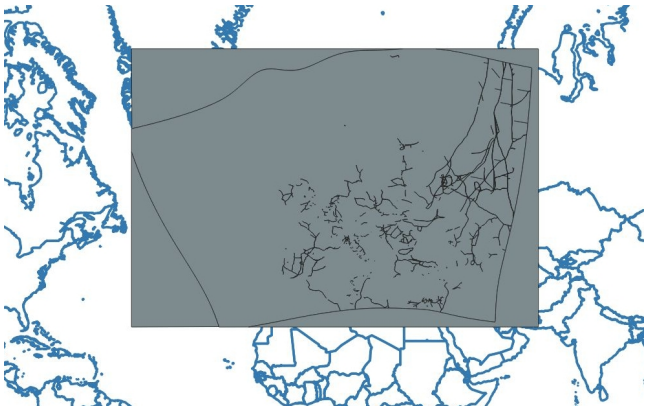


Figure 3.13: Sample shapefile, prior to clean up, where entire shapefile area is covered by one or several large polygons.

The goal is to remove all those polygons that are not lines, as is the case in Figure 3.14.

This can be achieved by using an application, such as QGIS, and selecting and removing the unwanted polygons. Figure 3.15 shows the entire shapefile, where a single polygon has been selected (yellow) which has been removed in the next process step, resulting in Figure 3.16.

As can be seen in Figure 3.17, even areas between pipelines can be polygons (grey area between pipelines). These need to be removed as well, and have been selected as shown in Figure 3.18, and results of the removal process can be found in Figure 3.19.

Figure 3.14: Sample shapefile, after the clean-up, where all polygons are pipelines.

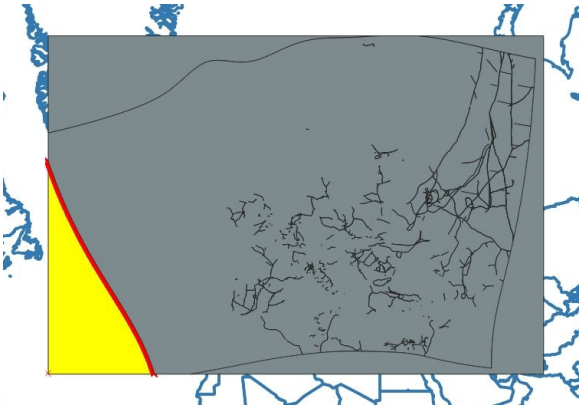


Figure 3.15: Sample shapefile, where a single polygon has been selected (yellow area with red stars).

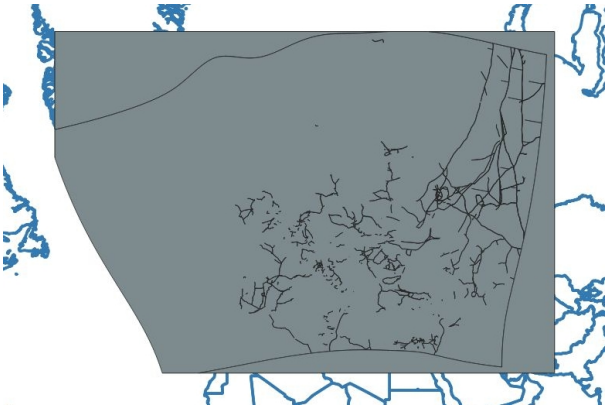


Figure 3.16: Sample shapefile, after the removal of the above selected polygon.

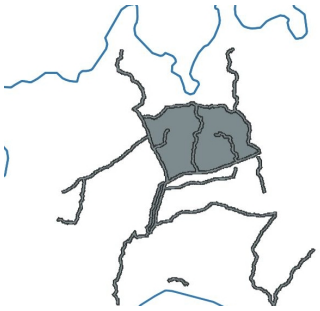


Figure 3.17: Sample shapefile, with polygon between pipelines.

Figure 3.18: Sample shapefile, with polygon selected between pipelines.

Figure 3.19: Sample shapefile, with above selected polygon removed.

In addition, there are polygons between parallel lines of pipelines, which need to be removed as well. Such a polygon between two parallel lines can be seen in Figure 3.20, which has been selected already (red). After the removal process (Figure 3.21) the two parallel lines are better visible and will make it easier for subsequent processes to carry out the conversion process from polygons to SciGRID_gas elements.

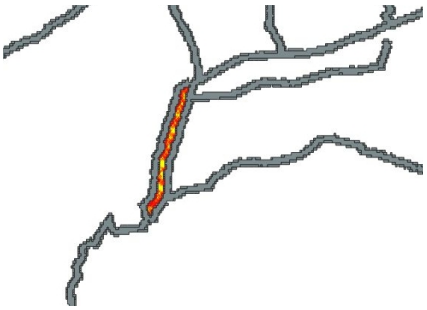


Figure 3.20: Sample shapefile, with polygon between two parallel pipelines selected (yellow and red).

Figure 3.21: Sample shapefile, with polygon between two parallel pipelines removed.

Resulting number of polygons per component group were large, most likely too large for a windows PC desk top, but small enough for the UNIX computer used. Hence this process was executed on a UNIX computer. Resulting files are written into the folder "../SciGRID_gas/Eingabe/Maps/EntsoG_2019/06_Polygon/One/".

Polygons to SciGRID_gas elements

The main function that is being used to convert the polygon files into SciGRID_gas elements is called **M_Maps.polygons2Netz()**. This function calls several functions from other modules, e.g. **GDP.GeoSerie**, or creates instances from other class definitions, e.g. **geometry.Centerline()**. For this process to work, component-specific parameters had to be determined, and will be part of the default settings. The outputs of this process for all of Europe generated a specific SciGRID_gas component data set.

Resulting data is being written into the folder "../SciGRID_gas/Eingabe/Maps/EntsoG_2019/07_A_CSV/One/".

Here the resulting pipeline data sets received an attribute called *pipe_class_EMap* (see Chapter 3.6.3).

Besides pipeline length, which will be generated dynamically at a later stage, this is the only attribute that was extracted from the PDF map.

Removing wrong elements

During the digitization process and the subsequent processes of converting the data into a SciGRID_gas data set, wrong lines started to appear in the data sets. These needed to be removed, as otherwise, they would be leading to wrong *PipeLines* elements, *Productions* sites, or *Storages* facilities. Hence, a function was written that removes *PipeLines* that are connected at only one end, and is called **M_Maps.multi_removeStichPipeLines()**. It was found that for some component elements, e.g. of type *PipeLines*, this function needed to be executed several times with varying settings, whereas it was not applied to any element of type *Productions*.

Resulting data is being written into the folder "../SciGRID_gas/Eingabe/Maps/EntsoG_2019/07_B_CSV/One/".

A further function was designed that removes *PipeLines* elements that are not connected at all. These are so called lone pipes and can be removed by the function **M_Maps.removeLonePipeLines()**. Here any lone pipelines shorter than 2.55 km were removed.

Resulting data is being written into the folder "../SciGRID_gas/Eingabe/Maps/EntsoG_2019/07_C_CSVs/One/".

Joining *PipeSegments*

At this part of the program, pipelines are being connected. However, this is being carried out for each group of pipelines, e.g. "PIPELINES > MEDIUM", or "PIPELINES > LARGE".

First of all, pipes were joined, if their end nodes were closer than a user defined distance. In a second step, all pipelines were broken up into smaller chunks ("chunking"), resulting in an increase of start and end nodes. Then it was investigated, if pipe ends were closer than a user specified value to any other pipe end. In case that the distance was shorter than the user specified value, new pipes were added connecting those nodes. In a subsequent step, the pipes were de-chunked again where possible. An example is given in Figure 3.22 and Figure 3.23.

The function carrying out this process is called **M_Maps.ExtraJoin()**, and resulting data is being written into the folder "../SciGRID_gas/Eingabe/Maps/EntsoG_2019/09_RawData/".

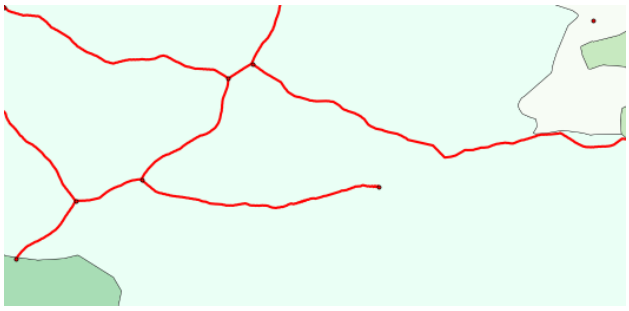


Figure 3.22: *PipeLines* in Belgium prior to chunking and joining.

Figure 3.23: *PipeLines* in Belgium joined through the chunking and joining process.

Generation of a single SciGRID_gas data set

In these subsequent steps, the different types of *PipeSegments* elements ("PIPELINES > MEDIUM", "PIPELINES > LARGE" etc.) are combined into a single SciGRID_gas data set. For this to happen, additional Python code needed to be executed to create additional connections, e.g. land-based pipelines and off-shore drilling platforms.

Hence, the function **M_Maps.joinDataSets()** joins all the separate data sets into a single SciGRID_gas data set. Resulting data is being written into the folder "../SciGRID_gas/Eingabe/Maps/EntsoG_2019/10_Final/". The method applied is similar to the one described under "Joining *PipeSegments*", however pipes were only allowed to be connected, if they were not of the same type, e.g. a "PIPELINES > MEDIUM" pipe was not connected with another "PIPELINES > MEDIUM" pipeline, but could be connected with a pipe of the group "PIPELINES > LARGE".

It was noticed that some vital pipelines were missing. Hence, an option has been implemented, so that additional connections (pipes) can be added. Information for the additional pipes is stored in a file called "../SciGRID_gas/Eingabe/Maps/EntsoG_2019/AdditionalPipeSegments.csv". It consists of a single header line, and the following five columns:

- long_1: Start longitude value of pipeline
- lat_1: Start latitude value of pipeline
- long_2: End longitude value of pipeline
- lat_2: End latitude value of pipeline
- Emap_Class: EMAP class value of pipeline.

Here the user can add as many pipelines as required. For the current data set the pipes added are given in Table 3.21.

Table 3.21: List of pipes added to EMAP data set.

long_1	lat_1	long_2	lat_2	Emap_Class
-1.9956	47.406	-2.45388	47.6069	2
9.5211	50.691	9.6987	50.7447	2
13.5415	50.6373	13.7921	50.5032	1
12.8496	48.26	13.2697	48.272	2
25.6503	59.4558	25.903	59.4199	3
28.0213	59.3778	28.1054	59.1055	3
24.6468	54.8711	25.0671	55.0209	2
31.3435	51.2878	31.5597	51.0152	2
31.0853	52.3925	31.393	52.0959	2
27.6117	57.8294	27.3961	57.6529	2

Last cosmetic alterations to single SciGRID_gas data set

The final function **M_Maps.finalTouch()** modifies the single SciGRID_gas data set, by carrying out the following actions:

- setting the country code according to the lat/long values
- setting elevation values according to the lat/long values.

Resulting data is being written into the folder "../SciGRID_gas/Eingabe/Maps/EntsoG_2019/15_Final/".

3.6.3 EMAP data density

For each component the data density for the most relevant attributes will be given next.

PipeSegments elements

Overall, there are 7126 *PipeSegments* elements in the resulting EMAP data set. Table 3.22 summarizes the data densities for the most important *PipeSegments* attributes:

Table 3.22: EMAP *PipeSegments* data density

Attribute name	Data density [%]
length_km	100
pipe_class_EMap	53
exact	100

As each element of the component *PipeSegments* originated from a line in the original PDF map, and as this map was geo-referenced, a length for each element could easily be determined, therefore the overall data density for the attribute *length_km* is 100 %. In addition, for the attribute *exact* a blanket value of 3 has been assumed, indicating that the topological accuracy would be better than 100 km (see Chapter 3.6.4).

The attribute *pipe_class_EMap* is a value that was generated during the data generation process. The original PDF file contained three different layers for three different pipeline thicknesses: "small", "medium", and "large". This was given for all pipelines in Europe, except for the regions of Germany and the North Sea. To be able to use that information of the "small", "medium" and "large" attribute during the heuristic processes, these attributes were converted into an integer number as given in Table 3.23. The German pipeline layer contained all pipelines, from small to large. Hence, an overall value of two was assumed. For the pipelines in the North Sea a mixture of small, medium and large pipelines was given. However, one can assume that due to the transport from production sites to country border points, the pipelines would be larger on average. Hence, a value of 1.5 has been assumed for all pipelines in the North Sea.

Table 3.23: EMAP *PipeSegments pipe_class_EMap* values

Type of pipe Segment	value
small	3
medium	2
large	1
Germany	2
North Sea	1.5

Storages elements

Overall, there are 177 *Storages* elements in the resulting EMAP data set.

The extraction process was not able to retrieve any further information for the *Storages*, except their locations.

Productions elements

Overall, there are 103 *Productions* sites in the resulting EMAP data set.

Again, the extraction process was not able to retrieve any other information for the *Productions*, other than their locations.

Nodes elements

Overall, there are 6040 *Nodes* elements in the resulting EMAP data set.

Table 3.24 summarizes the data densities for the most important *Nodes* attributes:

Table 3.24: EMAP *Nodes* data density

Attribute name	Data density [%]
exact	100
elevation_m	100

Here again, the information that has been used to generate a value for the attribute *exact* is the same as applied for the *PipeSegments*. Hence, each *Nodes* element was assigned a value of three.

The elevation attribute *elevation_m* was not retrieved from the EntsoG PDF map, but was generated using the APIs from Open Topo Data [Nis20] or Bing [Mic20].

3.6.4 Topological comparison with other data sets

As this is a non-OSM data set that has the potential of contributing a large volume of pipe data to this project, a brief topological comparison with other data sets has been carried out. This comparison focuses on the aspect of: "How good is the topological information in the EMAP data set?", or "What is the *exact* value for the EMAP data set? Should it be three or better?". To answer this question, the topology of the EMAP data set will be compared with the topology of the OSM data set. Here it is assumed that the OSM data set has the most correct topological values. Here a Hänsel und Gretel method will be used.

Hänsel und Gretel

The Hänsel und Gretel [AFW14] approach is a sampling-based distance approach, as it does take into consideration the topological flow of any pipeline, which is not the case for a Hausdorff distance [AG99].

The Hänsel und Gretel approach is based on placing a point (red "stone") along the pipelines every distance d > 0 (jump distance) for the first data set. This process is carried out for all pipelines of this first data set. In addition, the same is carried out for the other pipeline data set, using blue "stones". After this one looks for the closest distance between a red "stone" and any one of the blue "stones". One records this distance and removes those two "stones". After this, this distance finding is repeated, until one "stone" colour is gone. As one should realise, the distances found will increase, as the "stone" pairs of smaller distances have already been removed.

If one has two perfect pipeline networks, which are almost the same, e.g. the second data set has been derived by adding a small noise factor to the first one, then the distances between the coloured "stones" should be of the order of half of the jump distance (d/2) plus the small noise factor. However, as soon as one data set has fewer pipelines than the other, or the paths of the pipelines of the two data sets to be compared are quite topologically different, then a large portion of the distances found between the "stone" pairs will be larger than the jump distance. The raw OSM data set showed very good coverage for the country of Spain, hence the EMAP data set will be compared with the OSM data set for Spain (Figure 3.24). Here the emphasis is not regarding, which data set has a better coverage in pipelines, as a real representation of the pipelines is not known. However, the emphasis is on (with the assumption

that the OSM data set has correct topological information) how good is the topological information of the EMAP data set, as any distances between pipeline nodes (stones) larger than the jump distance will be due to the EAMP data set not representing the pipeline at the true locations.

Figure 3.24 depicts the pipelines of the OSM (black) and the EMAP (red) data sets for Spain.

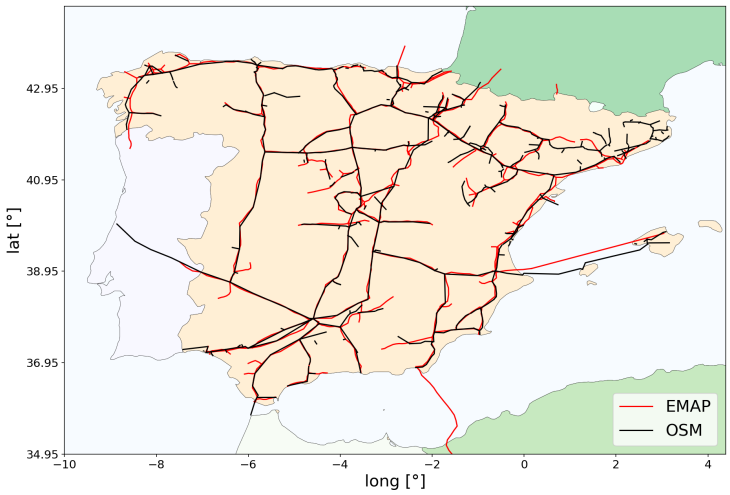


Figure 3.24: *PipeSegments* of the EMAP (red) and OSM (black) for Spain.

To be able to compare the "correct" pipelines, the data sets were adjusted accordingly:

1) The OSM (black) pipeline in Portugal was shortened, so that both data sets stopped at the border. Same was carried out for the EMAP (red) data set in the north of Portugal. (All pipes outside of the Spain were removed).

2) Pipes that were only in one data set, but not in the other were also removed, as the comparison is not about the completeness of the data set but only on the topological closeness. E.g. where the EMAP data set contained parallel lines, whereas the OSM data set contained only one line, one of the parallel lines from the EMAP data set waw removed. In addition, one data set might have a small pipeline leaving a main trunk, which is not present in the other data set, therefore this small pipeline was removed as well.

A jump distance has been set to 6 km. The smaller the jump distance the better. However, due to computer memory issues, any jump distance smaller than 6 km could not be computed on a standard Windows PC. However, this resulted in an OSM data set with 2097 *Nodes* (black "stones") and an EMAP data set of 1923 *Nodes* (red "stones"). In a perfect match, both data sets would have had the same number of *Nodes*. Hence, this is indicating that the OSM data set might follow a „wigglier" path when compared with the EAMP data set. However, it was also noticed that the OSM data set consisted of more *PipeSegments* (300) than the EMAP data set (256), and nodes will be placed wherever a pipe starts and ends, hence the OSM data set has more start and end nodes than the EMAP data set. Results of the data sets for Spain can be found in Figure 3.25.

In the next step, pairs of nodes from the different data sets were found, starting with the smallest distances first, and the cumulative number of pairs (in percent) found is depicted in respect of the distance in Figure 3.26.

A key feature of the result is the separation of 3 km (half the jump distance). For Spain this value is 58 %, meaning that 58 % of all node pairs have a separation of 3 km or less. This indicates that 58 % of the EMAP *PipeSegments* have the same topology as the OSM data set, and hence a "correct" topological value. In addition, it was found that 90 % of all EMAP pipes are within 14 km of the OSM data set, and this value increases to 95 % for a distance of 30 km. For any distance greater than this, the mismatch between incorrect nodes pairs would appear, where one node left of the OSM data set is to the West of Madrid, whereas the closest EMAP node can be found to the East of Madrid. Clearly, they are not from the same pipelines, which is also evident in the mismatch of node numbers of the two data sets (OSM consisted of 2097 *Nodes* and the EMAP data set consisted of 1923 *Nodes*).

So, to answer the above question "What is the *exact* value for the EMAP data set?", one knows that 88 % of the EMAP

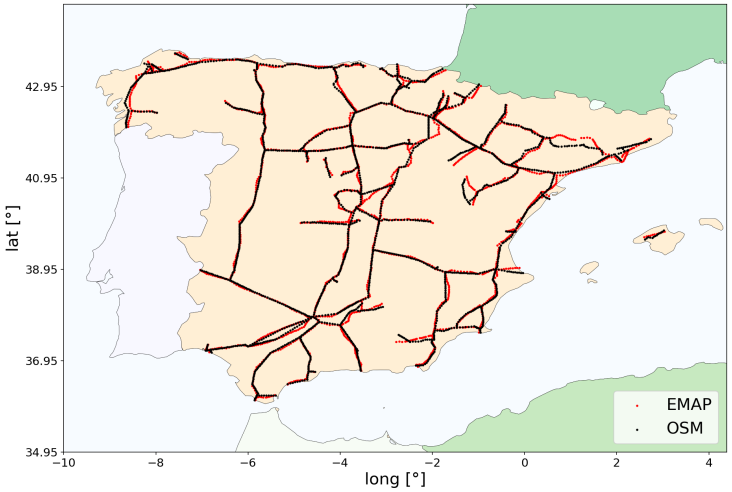


Figure 3.25: *Nodes* of the cleaned EMAP (red) and OSM (black) for Spain.

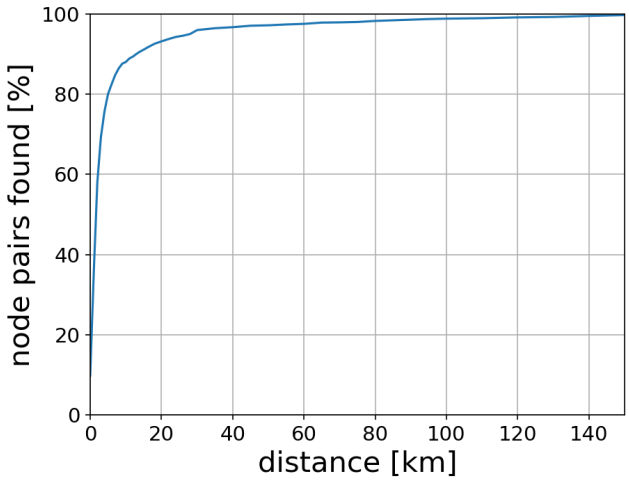


Figure 3.26: Cumulative Hänsel und Gretel results for Spain (ES).

data set is within 10 km of the OSM data set, whereas 12 % of the EMAP pipes is > 10 km from the "true" (OSM) pipes. With the definition of the attribute *exact* as specified in Chapter 10.3, the overall *exact* value for the EMAP data set is therefore being set to three (3).

3.6.5 Changes to previous releases

Changes in the code were implemented when compared with version 1. The code here followed a different pathway of generating and joining the individual data sets. The major difference here is that the problem of the pipes crossing borders has been eliminated.

3.6.6 Copyright

Copyright

Based on the legal framework, all of the EMAP was generated in such a way that it has a copyright that does not restrict us from making the data available to other users.

Hence, the following applies to the EMAP data:

Disclaimer

The EMAP data set is supplied on a best-effort basis only. While every effort is made to make sure the information is accurate and up-to-date, we do not accept any liability for any direct, indirect, or consequential loss or damage of any nature, however caused, which may be sustained as a result of reliance upon such information.

3.6.7 Summary EMAP Data

A PDF map of the European gas transmission network was available through the "EntsoG" transparency platform. Tools have been created to convert the PDF into the SciGRID_gas data structure and make the data accessible throughout the SciGRID_gas project.

As can be seen, the geo-referencing was sub-optimal for countries on the African continent. However, as the dataset that is required focuses on Europe (EU), those pipelines will not be part of the final SciGRID_gas data set. Having said this, their location is within the given certainty of 100 km (exact = 3).

The Table 3.25 summarises the number of elements for each component found:

Table 3.25: EMAP component element summary

Component Name	Count
Nodes	6040
PipeSegments	7126
Productions	103
Storages	177

The current version of the EMAP data set is presented for all of Europe in Figure 3.27, resulting in a total of 221,833 km of *PipeSegments* elements. However, it will need to be pointed out that a large fraction of those pipes is in countries, which are outside of the scope of this project. In addition, it will need to be pointed out that this process described here was not that good in determining parallel pipelines.

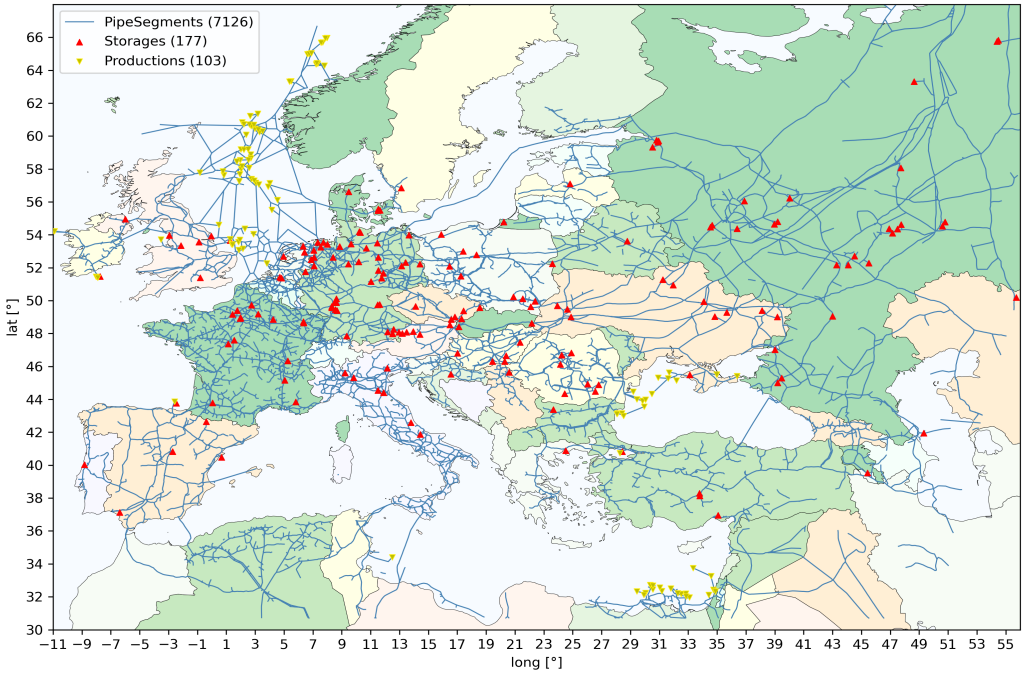


Figure 3.27: The pipelines, storage facilities and production sites of the EMAP data set.

3.7 The Long-term Planning and Short-term Optimization (LKD) data set

The **Long-term planning and short-term optimization** data set (**lk-DEU**) is the second of three non-OSM data sets that contain geo-referenced gas facilities. It was generated by several German research institutes and funded through the German government grants. It was part of a much larger research project (see link below). Here the gas facilities from the lk-DEU data set were used and incorporated into SciGRID_gas data model as the **LKD** data set. It contains information on gas pipelines, gas production sites, gas storages, compressor locations, and nodes.

As this data set is extremely well geo-referenced, it is of particular interest to the SciGRID_gas data project. The LKD data set can be used in conjunction with the OSM data set for training purposes, and as a data source for the heuristic processes, as a lot of attributes are available for a lot of elements. In addition, pipelines from the LKD data set can be copied into the final SciGRID_gas data set.

The LKD facilities data set came in form of a shapefiles, and consisted of polylines with some attributes, such as pipe diameter, max gas flow capacity and more. In addition, parts of the shapefiles were tables of facilities, with information on storages, production, and industrial demand. Great care was taken from the original data set producers, to create a data set with a vast number of attributes, which will be used throughout the SciGRID_gas project. Overall, the topological quality of the data set is good, as was verified by some sample checks. Gas sites could be found on satellite images within a few hundred meters. Due to the large number of elements, with a good selection of attributes and good topological information, the entire LKD data set has been incorporated into the SciGRID_gas project.

Further external information on the lk-DEU data set

More information on the data can be found under the following URL:

https://www.ewl.wiwi.uni-due.de/nl/forschung/forschungsprojekte-ewl/lkd-eu-langfristige-planung-und-kurzfristige-optimierung-des-elektrizitaetssystems-in-deutschland-im-europaeischen-kontext/.

This link describes the (Long-term planning and short-term optimization data sets of the German electricity system within the European) data set [KKS+17]. The project was a joined effort by:

- German Institute for Economic Research (DIW Berlin)
- Working group for Infrastructure Policy (WIP) at Technische Universitaet Berlin (TUB)
- Chair of Energy Economics (EE2) at Technische Universitaet Dresden (TUD)
- House of Energy Markets & Finance at University of Duisburg-Essen.

This project was funded by the German Federal Ministry for Economic Affairs and Energy through the grant "LKD-EU", FKZ 03ET4028A, with the aim of presenting a status quo of the German energy sector. The following three energy media were part of their project:

- electricity
- heat
- natural gas.

Here only the gas components are being used.

3.7.1 Pre-requirements for accessing the LKD data set

The SciGRID_gas project has received the right to use, change and redistribute the LKD data under an open license agreement. However, if you use this data or any data set which incorporates this data, you are also required to cite the original authors of the LKD data as follows:

Kunz et al. 2017, Data Documentation: Electricity, Heat, and Gas Sector Data for Modeling the German System

In addition, the data set can be downloaded from the following location:

https://zenodo.org/record/1044463#.Xah7i2ZCSUk

Please put a copy into the following location:

/SciGRID_gas/Eingabe/LKD/

In addition, some data changes needed to be carried out due to small mistakes in the LKD data. However, these have been carried out by the SciGRID_gas project and corrected data has been written to the CSV LKD output data set. The SciGRID_gas tools that carry out those changes will be supplied as part of the SciGRID_gas project.

3.7.2 Data processing of the LKD data

The LKD gas facilities data set came in the form of several shapefiles. It contained several tables, which were read in with tools and dissected to fit into the data structure of the SciGRID_gas project. The components that were read in are:

- *PipeSegments*, from the 'pipelines_utf8.shp' shapefile
- *Nodes*, from the 'nodes_utf8.shp' shapefile
- *Productions*, from the 'productions_utf8.shp' shapefile
- *Storages*, from the 'storages_utf8.shp' shapefile.

Subsequent to reading the data from the shapefiles, it was necessary to convert the data so that it adheres to the SciGRID_gas data structure. Some inconsistencies were found with the data set. The following fixes of the LKD data set had to be carried out:

- Some node ids were found more than once in the original data set for different nodes. Hence, this was manually rectified by changing node ids for 29 nodes.
- Some nodes had a wrong country code setting. For 10 nodes the country code attribute needed to be changed.

In an additional step, the elements of type *Compressors* were generated by using information that was supplied with the *Nodes*. The Node elements contained an attribute "comp_units", which stands for "number of compressor units". Hence, if this value was larger than 0, then it was assumed that the node contained a single compressor element at that location. In addition, the attribute "comp_units" was then used as the value for the number of compressor turbines at the compressor location. E.g. if the value was two, then the compressor element's attribute *num_turb* was set to two.

At this stage of the LKD data process, there were more than 1800 pipe-segments with more than 1400 nodes. It was not apparent why there were so many nodes and pipe-segments. For many pipe-segments, two individual pipe-segments that connect with the same node contained the same attributes with the same values, and the node in question only connected two pipe-segments, not forming a T-section. Hence, pipe-segments were joined and nodes removed if the following rules applied:

- The node in question connects only two pipe-segments.
- The attributes values for *max_pressure_bar*, *is_H_gas*, *diameter_mm* and *pipe_class_LKD* needed to be identical for both pipe-segments. An exception is made for the node "Haidach" and "N_805129", where no pipe-segment joining took place.

In addition, the following simplifications of the network were carried out:

- Nodes that were closer than 3 km were merged, removing some pipelines
- Pipelines that were connected to only one other element (pipeline or non-pipe component) were removed, if they were shorter than 5 km.

These processes reduced the number of segments to 1085, and the overall number of nodes to 721.

For some of the attribute values, the unit of the attribute value did not "agree" with the units used within the SciGRID_gas data project. Hence, unit transformation (see Chapter 10.2) had to be carried out for the following attributes of the following components:

- "Storages", attribute converted from *max_cap_pipe2store_GWh_per_d* to *max_cap_pipe2store_M_m3_per_d*
- "Storages", attribute converted from *max_cap_store2pipe_GWh_per_d* to *max_cap_store2pipe_M_m3_per_d*.

Subsequently, the old attributes with the "wrong" units were removed from the component data set.

Further attributes were added to the component:

- The length of the pipe-segment was derived using the polylines of each pipe-segment.
- The average latitude and average longitude were calculated by using the polylines of each pipe-segment.

In addition, the attribute "exact" was added to each *Nodes* element and a value of one was given.

In addition, for each *Nodes* element the following attributes were removed:

- 'compressor'
- 'ugs'
- 'production'
- 'comp_units'.

3.7.3 Further alterations to the LKD data set

Estimation of the attribute *max_cap_GWh_per_d*

The original data set contained for some pipelines the attribute *max_cap_GWh_per_d*. However, it was found that this value was incorrect. Therefore the attribute value *max_cap_M_m3_per_d* was generated, as described in [KKS+17], Chapter 4.2.2. To achieve this, a heuristic relationship was formed where the following were the independent variables: *max_pressure_bar*, *diameter_mm*, and *pipe_class_LKD*, and the attribute *max_cap_GWh_per_d* is the dependent attribute variable. The backbone of the heuristic relationship is the information from Table 25 from [KKS+17]. Here it is assumed that all values given in the original data set of the attributes *max_pressure_bar*, *diameter_mm*, and *pipe_class_LKD* have the same quality, no matter if found or estimated, and can be used in this heuristic process. This process generated a *max_cap_GWh_per_d* value for 937 pipesegments.

3.7.4 LKD data density

The data of the LKD data set contains elements from the following components:

- *PipeSegments*
- *Compressors*
- *Productions*
- *Nodes*
- *Storages*.

Each of those components and their attributes will be described below.

As all components have the following attributes, they are presented here once:

- *id*: unique identifier
- *name*: name of the pipe-segment
- *source_id*: a source id
- *node_id*: the id of the start and the end node of the pipe-segment
- *lat*: a list of latitude values
- *longitude*: a list of longitude values
- *country_code*: a string pair indicating the country code of the start and the end point
- *comment*: a user comment.

PipeSegments elements

Overall, there are 1085 *PipeSegments* elements in the LKD data set. In addition to the default attributes, the following non-standard attributes (see Table 3.26) are supplied. The number of attribute values supplied for each attribute is given by the column 'Data density [%]':

Table 3.26: LKD *PipeSegments* data summary

Attribute name	Description	Units	Data density [%]
diameter_mm	a pipe diameter	mm	88
is_H_gas	the gas type identifier	1 or 0	100
length_km	the total distance of the pipe-segment	km	100
max_cap_M_m3_per_d	maximum gas flow capacity	$\mathrm{Mm^3 d^{-1}}$	86
max_pressure_bar	maximum allowed pressure in the gas pipe	bar	83
operator_name	operator name		99
pipe_class_LKD	gas pipe-segment class type	1 to 6	87
lat_mean	calculated mean latitude value	degree	100
long_mean	calculated mean longitude value	degree	100

pipe_class_LKD

For reasons of attribute generation at a later stage, the values for *pipe_class_LKD* have been converted from A, B, C,... to 1, 2, 3,..

Compressors elements

Overall, there are 13 *Compressors* elements in the LKD data set. In addition to the default attributes, the following non-standard attributes were supplied (see Table 3.27) and partially populated for the component *Compressors*.

Table 3.27: LKD *Compressors* data summary

Attribute name	Description	Units	Data density [%]
entsog_key	key associated with EntsoG facility		38
license	indicator of the license		100
num_turb	the number of compressor turbines		100
operator_name	name of the operator		100

Storages elements

Overall, there are 14 *Storages* elements in the LKD data set. In addition to the default attributes, the following non-standard attributes (see Table 3.28) are supplied and populated for the component *Storages*.

Table 3.28: LKD *Storages* data summary

Attribute name	Description	Units	Data density [%]
entsog_key	key associated with EntsoG facility		100
max_cap_pipe2store_M_m3_per_d	maximum gas flow from the network into the storage unit	$\mathrm{Mm^3 d^{-1}}$	100
max_cap_store2pipe_M_m3_per_d	maximum gas flow from the storage unit into the network	$\mathrm{Mm^3 d^{-1}}$	100
operator_name	name of the operator		100

Productions elements

Overall, there are 6 *Productions* elements in the LKD data set. In addition to the default attributes, the following non-standard attributes (see Table 3.29) were supplied and populated for the component *Productions*.

Table 3.29: LKD *Productions* data summary

Attribute name	Description	Units	Data density [%]
entsog_key	key associated with EntsoG facility		100
max_supply_M_m3_per_d	maximum gas production	$\mathrm{Mm^3 d^{-1}}$	100
is_H_gas	boolean indicating that H gas type	1 or 0	100
operator_name	name of the operator		100

Nodes elements

Overall, there are 721 *Nodes* elements in the LKD data set. In addition to the default attributes, the following non-standard attributes (see Table 3.30) are supplied and partially populated for the component *Nodes*.

Table 3.30: LKD *Nodes* data summary

Attribute name	Description	Units	Data density [%]
crossborder	boolean indicating that node is a gas cross border point		98
entry	boolean indicating that node is a gas entry point		98
entsog_key	key associated with EntsoG facility		9
exact	value indicating the accuracy in geo-referencing	1 to 5	100
exit	boolean indicating that node is a gas exit point		98
H_L_conver	boolean indicating if converter between H & L gas		98
license	license key		98
operator_Z	additional operator name		64
operator_name	name of the operator		96

In addition the NUTS values (Nomenclature des unités territoriales statistiques) [Wik21] have been added to the nodes for NUTS-1, NUTS-2 and NUTS-3. For this the latitude and longitude of each node was used in a look-up in a spatial shape file. Hence for those values, the data density is 100 %.

Additional data from the LKD data set

In addition, there is data on gas demand on a spatial level of NUTS-3, leading to 402 elements. However, currently this is not being used, but might be used at a later stage.

3.7.5 Copyright and disclaimer for the LKD data set

The lk-DEU data set has been published under the Creative Commons Attribution 4.0 International Public License. This allows us to use the data in this project and re-distribute the data as well.

Disclaimer

The LKD data set is supplied on a best-effort basis only. While every effort is made to make sure the information is accurate and up-to-date, we do not accept any liability for any direct, indirect, or consequential loss or damage of any nature, however caused, which may be sustained as a result of reliance upon such information.

Acknowledgement

We would like to acknowledge the "Deutsches Institut für Wirtschaftsforschung" (Mohrenstr. 58, 10117 Berlin, Germany) for allowing the SciGRID_gas project to use their data.

3.7.6 Summary LKD data

The gas pipeline and gas facilities from the LKD data set is of great importance to the SciGRID_gas project. It is one of only three non-OSM data sets that contain gas facilities that are geo-referenced, and hence, can be used for validation processes covering all of Germany. In addition, it contains some attribute values in respect of gas pipelines that are fundamental for the gas data model. This data set was made available through a German research project and downloadable from the project's web page. Tools have been written to load the LKD shapefiles and make them accessible for the SciGRID_gas project.

Below Table 3.31 summarises the number of elements for each component found:

Table 3.31: LKD component summary

Component Name	Count
Compressors	13
Nodes	721
PipeSegments	1085
Productions	6
Storages	14

In addition, the map in Figure 3.28 visualizes the data for Germany, of its more than 27,000 km transmission network.

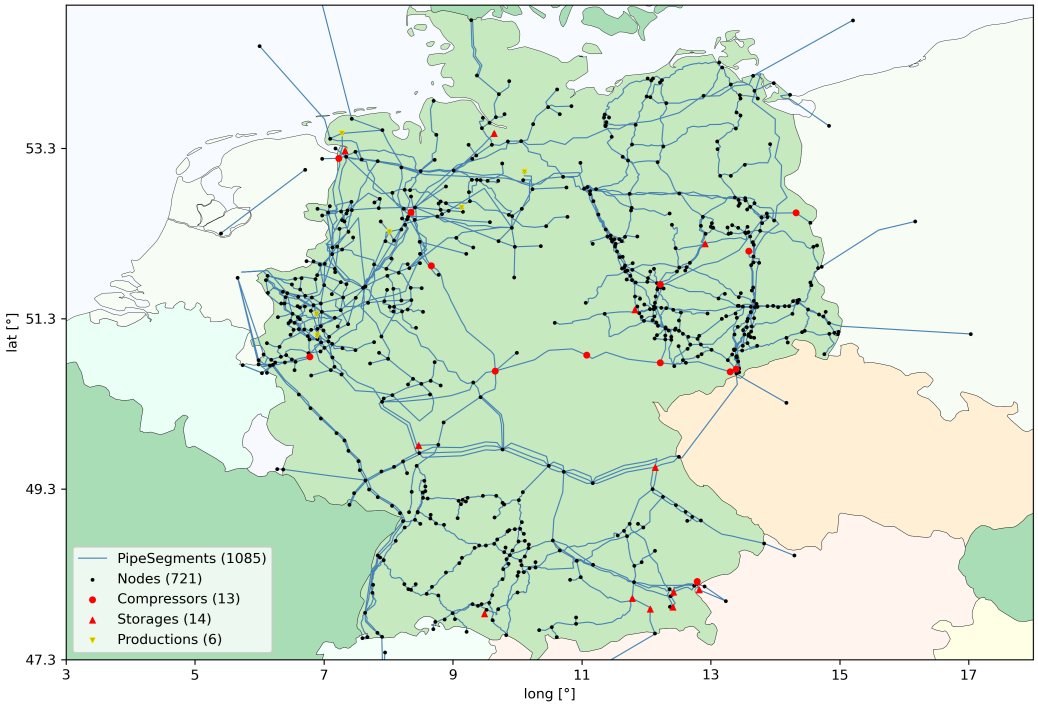


Figure 3.28: Map of components of the LKD data set.

3.8 The Great Britain (GB) data set

"National**grid** [nationalGrid20]" is the main TSO for Great Britain, covering the electricity and the gas networks. The networks cover England, Scotland and Wales. The project SciGRID_gas is very fortunate that national**grid** allows for the download of the geo-referenced gas facilities data and gas time series data. The facility data can be found under: https://www.nationalgridgas.com/document/81201/download. In addition, instantaneous gas flow time series data can be downloaded from the following site: https://mip-prod-web.azurewebsites.net/userdefineddownload. As the data originates from Great Britain, the abbreviation used throughout this document for this data set is **GB**.

The facilities data set came in form of a shapefile, and consists of polylines with some attributes, such as pipe diameter. In addition, part of the shapefile was a table of facilities, with further information, such as an entry indicating if the facility is a compressor or not. Overall, the topological quality of the data set is **very** high, as by doing sample checks, all gas sites could be found on satellite images. Due to its "power", the entire spatial data set of national**grid** is being loaded by tools, and has been used for creating a better SciGRID_gas data set.

Currently, the temporal gas flow data is not being utilised. However, at a later stage, one could download the time series information and derive additional metadata, such as pipeline capacity, gas flow direction, and others.

3.8.1 Pre-requirements for accessing the GB data set

The GB data set is a further data set containing information that should be incorporated into the SciGRID_gas data set, as it contains detailed information on location of pipelines and other facilities throughout England, Scotland, and Wales. However, the copyright of this data set again is more restrictive than the copyright of the SciGRID_gas data project, therefore the SciGRID_gas project is not allowed to pass on the GB data set. However, you can download this data set yourself and use the SciGRID_gas project's code to incorporate the data. The data can be found through the following page: https://www.nationalgridgas.com/land-and-assets/network-route-maps#tab-1.

This data set comes as a Zip file and needs to be unpacked into the following location on your computer:

/SciGRID_gas/Eingabe/GB/

The following files can be removed, as they contain information for the electricity grid lines:

- Cable.*
- OHL.*
- Towers.*.

3.8.2 Data processing of the GB data

The gas facilities data set for the UK came in form of a shapefile. It contained several tables, which were read in with tools generated as part of the SciGRID_gas project and converted to fit into the data structure of SciGRID_gas. The components that were read in were:

- *PipeLines*, from the Gas_Pipe data set
- *Compressors*, from the Gas_Site data set
- *InterConnection* points, from the Gas_Site data set
- *Nodes*, from the Gas_Pipe data set.

Subsequent to reading the data from the shapefiles, it was necessary to process the data so that it adheres to the SciGRID_gas data structure. As the data for the UK was supplied in a different spatial projection, all lat/long values had to be converted from "epsg:27700" to "epsg:4326". In addition, the nodes lost the attribute "Compressors", as compressors are its own component within SciGRID_gas. Pipes of type *PipeLines* were converted to pipes of type

PipeSegments. Further for element of type *PipeSegments*, additional attributes "lat_mean" and "long_mean" were calculated and added.

All these changes made the UK data set compatible with the SciGRID_gas data structure.

Below specific steps taken for individual components are given.

Compressor data pre-processing

The compressor data is read in from the shapefile "Gas_Pipe.shp". As that file also contains information for non-compressor sites, a filter was used to select those entries, which had **COMP** in the field *SITE_TYPE*. Due to the information supplied, only locations (lat/long) for compressors are known. Other information of interest, such as number of turbines, type and power of turbines, were not given through the data set. Compressors were supplied as a poly-shape from the file "Gas_Pipe.shp". Hence, during the loading process, a single lat/long center value was determined for each of those poly-shapes.

After all, there were 21 *Compressors* elements generated. However, no other attribute values, such as capacity or max pressure were available for this component.

Connection point data pre-processing

The *ConnectionPoints* data is read in from the shapefile "Gas_Site.shp". As that file also contains information for compressor sites, a filter was used to select those entries, which had **AGI** of **TCSITE** in the field *SITE_TYPE*. Only location information is known for *ConnectionPoints*. *ConnectionPoints* were supplied as a poly-shape from the file "Gas_Pipe.shp". Hence, during the loading process, a single lat/long center value was determined for each of those poly-shapes.

After all, there were 147 *ConnectionPoints* elements in the GB data set. However, no other attribute values, such as max capacity or max pressure were available for this component.

Pipeline data processing

There were 291 polylines in the shapefile "Gas_Pipe.shp". However, there were a lot more pipelines to be seen on the map. This is because most polylines contained several parts, such as T-junctions, but also several *PipeSegments*. Hence, part of the loading process is to convert each polyline part into individual *PipeLines* elements. In addition, there was access to additional sites, such as compressors and connection points, from the "Gas_Site.shp" shapefile. Those two data sets needed connecting/combining as well. Hence, a further process while creating the pipelines was to join the geographic coordinates of pipelines with those of compressors and other facilities.

To split up the polylines and connect them to existing *Compressors* and *ConnectionPoints* elements, the following steps were carried out:

- Reading in a polyline.
- Determine number of parts to the polyline.
- If a polyline consisted of only one part, then the entire polyline was converted into a single *PipeLines* element.
- If a polyline consisted of more than one part, each of those parts is converted into a single element of type *PipeLines*.
- Converting a *PipeLines* element into a *PipeSegments* element.

This process generated 386 elements of type *PipeSegments*.

While generating the elements of type *PipeSegments* it was noticed that a large number of different nodes had the same latitude and longitude values. Hence, nodes with the same pair of latitude and longitude values were merged into a single node element, reducing the number of nodes by about 350 nodes. Further it was noticed that some nodes

were not used by any gas element. Hence, those were removed as well, leading to a further reduction by more than 200 nodes. In a next step it was found that some nodes were quite close together. Hence, a radius of 2.5 km was selected, where nodes that fell within this radius were merged into a single node, resulting in a further reduction of nodes by more than 460 nodes. This needed to be implemented, as pipes around facilities such as compressors were not connected, but needed to be connected. Further analysis revealed that some individual pipes started and finished at the same node, and all of those pipes had a physical length of less than 6 km. It is assumed that those pipes are not loops, for linepack[1], but were generated by the implemented processes above, and hence, needed to be removed. This led to a reduction of *PipeSegments* elements down to 340 elements.

Elements of type *Compressors* and *ConnectionPoints* are additional *Nodes*. However, only a center lat/long value was stored for each of those. These lat/long value pairs were not part of the pipe lat/long pairs. To make the lat/long pairs of the *Compressors* and *ConnectionPoints* elements part of the *PipeSegments*, the following steps were taken:

- For each element of type *Compressors*/*ConnectionPoints* their center lat/long value was selected.
- These lat/long value pairs were compared with all lat/long value pairs of the *PipeSegments*, and the one with the shortest separation selected.
- If above separation is smaller than a user specified distance, then the element of type *Compressors*/*ConnectionPoints* was "merged" with the pipe. The "merger" for type *Compressors*/*ConnectionPoints* is achieved by:
  - Changing the lat/long value of the selected pipe lat/long pair.
  - As an element of type *Compressors*/*ConnectionPoints* is termed to be a node as well, the pipeline was split at the point of the *Compressors*/*ConnectionPoints* element, hence, increasing the number of *PipeSegments*.

This process increases the number of *PipeSegments* to 355.

Nodes data

Nodes are the locations for all of the following elements:

- Start and end points of *PipeSegments*
- *Compressors*
- *ConnectionPoints*.

After all, there are about 292 nodes throughout England, Scotland and Wales.

3.8.3 GB data density

The data of the GB data set contains the following components:

- *PipeSegments*
- *Compressors*
- *ConnectionPoints*
- *Nodes*.

Each component is derived from shapefile tables. Below you will find a summary of the information as it will appear after the conversion to the SciGRID_gas data format.

As all components have the following attributes:

- *id*: unique identifier

[1] "Linepack" is a term used that refers to the volume of gas in a line being increased to store gas.

- *name*: name of the *PipeSegments* element
- *source_id*: a source id
- *node_id*: the id of the start and the end node of the pipe-segment
- *lat*: a list of latitude values
- *longitude*: a list of longitude values
- *country_code*: a string pair indicating the country code of the start and the end point
- *comment*: a user comment.

All additional attributes are stored in the *param* dictionary.

Compressors elements

As *Compressors* elements were derived from the pipeline shapefile, where only the number of *Compressors* elements was given for a location, no further information or attribute values were known for these elements. The only information given is their geo-reference location, which is very accurate.

Overall, there are 21 *Compressors* facilities in the GB data set.

ConnectionPoints elements

As connection points were also derived from the "Gas_Site.shp" shapefile, there was no further data for those connection points. Hence, the only information given is their geo-reference location, which is very accurate.

Overall, there are 147 *ConnectionPoints* elements in the GB data set.

PipeSegments elements

PipeSegments elements were derived from pipeline information which was read in from the "Gas_Pipe.shp" shapefile. The only additional metadata that was available for pipelines was their diameter. In addition, pipelines contained the attribute length. After the conversion of *PipeLines* to *PipeSegments*, the length was re-calculated for each element, based on the original path of each element. In addition, a mean latitude and longitude value was calculated and added as an attribute value to the *PipeSegments*. In addition, it is assumed that the UK transports only high calorific gas, hence, the attribute *is_H_gas* was added and set to a value of one for each element.

Overall, there are 386 *PipeSegments* elements in the GB data set. In addition to the default attributes, the following non-standard attributes (see Table 3.32) are supplied and partially populated with data.

Table 3.32: GB *PipeSegments* data summary

Attribute name	Description	Units	Data density [%]
exact	accuracy in geo-referencing		100
diameter_mm	a pipe diameter	mm	96
is_H_gas	boolean indicating that gas is of type H		100
lat_mean	calculated mean latitude value	degree	100
long_mean	calculated mean longitude value	degree	100
length_km	the total distance of the *PipeSegments*	km	100

Nodes elements

Nodes elements were derived from all other GB data components. Here it was assured that the same locations (same latitude and longitude) did not appear more than once in the data set.

Overall, there are 297 *Nodes* elements in the GB data set. Next to the default attributes, the attribute *exact* could be derived from the original data set, and hence, will be given for each node. As the original data source is in form of a shape file, and it was tested with satellite images at several locations and therefore it can be assumed that all nodes have an exact value of "1".

3.8.4 Copyright and disclaimer for the GB data set

Data availability and data usage

The data is provided by the "national**grid**" operator of the UK for the National Transmission System (NTS), including real time flow data and the latest operational news, sampled every two minutes. Here we only use a set of shapefiles, which contains geo-referenced gas pipelines, compressors and connection points.

Real-time data could be downloaded from the following link. However, this is currently not implemented. https://mip-prod-web.azurewebsites.net/userdefineddownload

Copyright

The copyright regulations of this data can be found under (https://www.nationalgridgas.com/land-and-assets/network-route-maps) and is given as:

> "These data sets are for indicative purposes only. They can only be used for emergency and land use planning and cannot be used for commercial purposes. They are owned by National Grid and you are required to acknowledge us in your product or application using "© National Grid UK".

Data disclaimer

In addition the data disclaimer is given as (https://www.nationalgridgas.com/land-and-assets/network-route-maps):

> "This data is supplied on a best-effort basis only, using available information as documented at the time by the transmission network operators. While every effort is made to make sure the information is accurate and up-to-date, we do not accept any liability for any direct, indirect, or consequential loss or damage of any nature–However, caused–which may be sustained as a result of reliance upon such information."

Acknowledgement

We acknowledge "© National Grid UK" as the original owners and creators of the raw data.

3.8.5 Summary GB data

The gas pipeline and gas facility from the GB data set is of great importance to the SciGRID_gas project. It is one of only three non-OSM data sets that contain gas facilities that are geo-referenced, and hence, can be used for validation processes, for England, Wales and Scotland. In addition, it contains some attribute values in respect of gas pipelines that are fundamental for the gas data model. This data set was available through the internet and was downloadable from the UK "national**grid**" operator. Tools have been written to load the GB shapefiles and make them accessible throughout the SciGRID_gas project.

Table 3.33 summarises the number of elements for each component found:

Table 3.33: GB component summary

Component Name	Count
Compressors	21
ConnectionPoints	147
Nodes	297
PipeSegments	386

In addition, a map (see Figure 3.29) visualizes the more than 8,000 km gas transmission dataset for the UK.

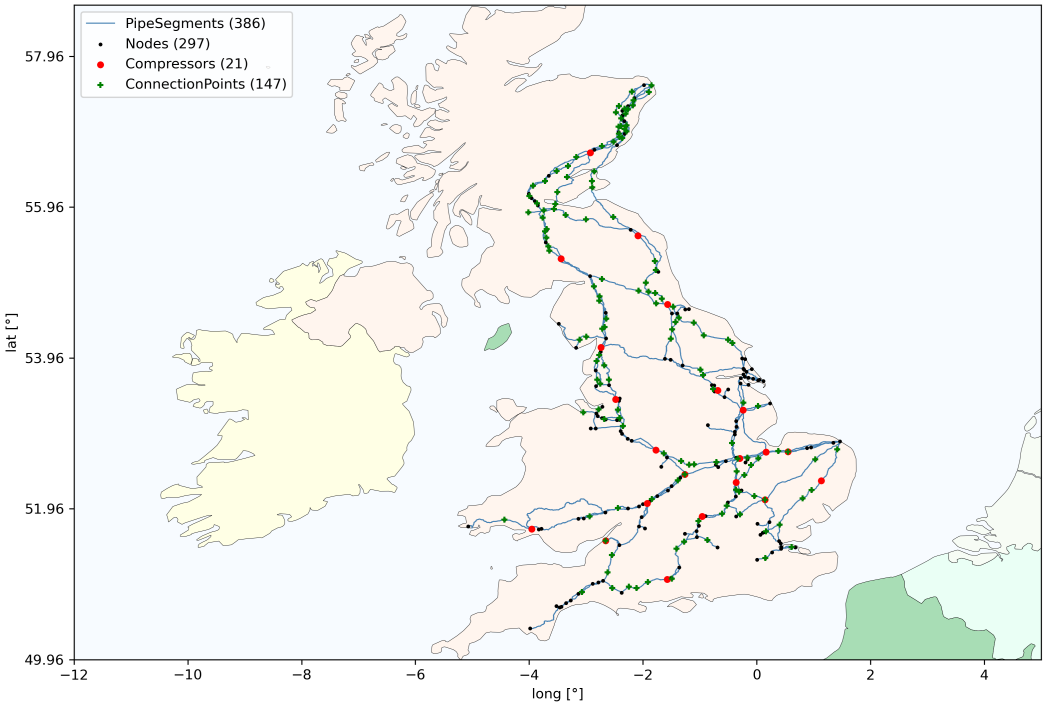


Figure 3.29: Map of components of the GB data set.

3.9 The Norway (NO) data set

Norway has one main national lines operator, being **Gassco** [Gassco20a][Gassco20b]. It covers the Norwegian continental shelf for the energy sources of gas and oil and connects Continental Europe and Great Britain. **Gassco** allows for the download of the geo-referenced non-infield gas and oil facilities data through the **Norwegian Petroleum Directorate**. The facility data can be found under: https://www.npd.no/en/about-us/information-services/available-data/map-services/. The file to download from the table with all those links is the entry for "TUF" ("Main pipelines. The dataset contains not infield pipelines."). The data covers the territorial waters of Norway, France, Great Britain, Germany, Denmark, the Netherlands and Belgium, and will be referred to as the Norway data set (**NO**).

The facilities data set comes in form of a shapefile, and consists of polylines with important attributes, such as pipe diameter. Overall, the topological quality of the data set is very high. The entire spatial data set is automatically incorporated into the SciGRID_gas data project.

Additional meta data has been made available in the form of an Excel book. Among others, it contains vital pipeline capacity information. Hence, this Excel data set needs to be merged with the Norwegian pipelines data set. The Excel data set can be found under the following URL:

https://www.norskpetroleum.no/en/production-and-exports/the-oil-and-gas-pipeline-system/#gas-pipelines

There is a "Download data" button above the "Gas pipelines on the Norwegian continental shelf" table, which will get the user the additional meta data file. This Excel table needs to be downloaded and converted to a CSV file, by saving the main sheet from that XLSX file. The location of this file shall be the same as where the shapefiles have been stored to, and shall have the file name "NorwayMetaInfo_01.CSV".

3.9.1 Data processing of the Norway data

The gas facilities data set for Norway came in form of shapefiles. It contains a table, which was read in with SciGRID_gas tools and converted to fit into the SciGRID_gas data structure. The components that were read in are:

- *PipeLines*
- *Nodes*.

Subsequent to reading the data from the shapefiles, it was necessary to process the data so that it adheres to the SciGRID_gas data structure. As the data for Norway was supplied in a different spatial projection, all lat/long values had to be converted from "epsg:4230" to "epsg:4326". Pipelines were converted to gas elements of component *PipeSegments*. Further, for each *PipeSegments* element, additional attributes "lat_mean" and "long_mean" were calculated and added.

All these changes made the Norway data set compatible with the SciGRID_gas data structure.

Below specific steps taken for individual components were given.

In addition, the meta data CSV file ("NorwayMetaInfo_01.CSV") was also read in, and linked via the location names that were present in both data sets. This allowed for the pipe segments to also contain information on the gas flow capacities.

Pipeline data processing

There were 70 polylines in the shapefile "pipLine.shp". However, there were a lot more pipelines to be seen on the corresponding map, with a tool like QGIS. This is because an individual polyline can contain several parts, such as T-junctions and several pipe-segments. Hence, part of the Python loading process was to convert all polyline parts into individual *PipeSegments* elements. Hence, the following steps were carried out:

- Reading in a polyline
- Determine number of parts to the polyline
- If a polyline consisted of only one part, then the entire polyline was converted into a single *PipeSegments* element
- If a polyline consisted of more than one part, then the following steps were carried out:
  - Information where new parts of a *PipeSegments* element started within a polyline was given through the variable *parts*.
  - New *PipeSegments* elements start at integer values supplied through *parts*.
  - Polyline parts were converted into *PipeSegments* elements.

This process generated 43 *PipeSegments* elements.

Nodes data

It was also possible to generate *Nodes* elements from the above information. *Nodes* are the locations for start and end points of *PipeSegments* elements.

After all, there are 53 *Nodes* elements throughout the territorial waters of Norway, France, Great Britain, Germany, Denmark, the Netherlands and Belgium.

The attribute *elevation_m* was not given through the original data set. However an attribute *waterDepth_m* was given for each pipe segment. Hence the attributer value *elevation_m* for each node was generated by averaging over all attribute values *waterDepth_m* from those pipes that are connected with the node. The corresponding uncertainty for *elevation_m* is the mean absolute difference of the input data set for each node. in case that the node was connected to only one pipe segment, then the uncertainty was set to 10 % of the pipe water depth value. With this approach, all *Nodes* element could be assigned an attribute value for *elevation_m*.

3.9.2 NO data density

The data of the Norway data set contained the following components:

- *PipeSegments*
- *Nodes*.

Each component is derived from a shapefile table. Below a summary of the information is given, as the data will appear after the conversion process into the SciGRID_gas data format.

PipeSegments elements

PipeSegments elements were derived from pipeline information which was read in from the "pipLine.shp" shapefile. The only additional metadata that was available for *PipeLines* element was their diameter. After the conversion of *PipeLines* element to *PipeSegments* element, the length was calculated for each pipe-segment, based on the polyline values for each *PipeSegments* element. In addition, a mean latitude and longitude value (*lat_mean*, *long_mean*) was calculated and added as an attribute value to the *PipeSegments* element.

Overall, there are 43 *PipeSegments* elements in the Norway data set. The *PipeSegments* elements have the following mandatory attributes:

- *id*: unique identifier
- *name*: name of the pipe-segment
- *source_id*: a source id
- *node_id*: the id of the start and the end node of the pipe-segment
- *lat*: a list of latitude values
- *long*: a list of longitude values
- *country_code*: a string pair indicating the country code of the start and end points
- *comment*: a user comment.

In addition, the following non-standard attributes are supplied (see Table 3.34) and are also given in respect of their data density (see Chapter 10.1 for a definition of 'data density'):

Table 3.34: NO *PipeSegments* data density summary

Attribute name	Description	Units	data density [%]
diameter_mm	a pipe diameter	mm	100
max_cap_M_m3_per_d	daily gas flow capacity	$\mathrm{Mm^3 d^{-1}}$	100
waterDepth_m	depth of pipeline	m	100
is_H_gas	boolean if gas is high calorific		100

Nodes elements

Nodes were derived from all other Norway data components. Here it was assured that the same locations (same latitude and longitude) did not appear more than once in the data set. In addition, they have the attribute of *country_code* which is set to "NO" for all of them. In addition, each node received an attribute "exact" with a value of one. Overall there were 53 *Nodes* elements found in the Norwegian data set. The attribute *elevation_m* was added to each element and populated through a look up through the Bing web page [Mic20].

Overall, there are 53 node elements in the Norway data set. The node elements have the following mandatory attributes:

- *id*: unique identifier
- *name*: name of the pipe-segment
- *source_id*: a source id
- *node_id*: the node id of the location of the compressor
- *lat*: a latitude value
- *long*: a longitude value
- *country_code*: a string indicating the country code of the compressor location

- *comment*: a user comment.

In addition, the following non-standard attributes are supplied (see Table 3.35) and partially populated for *Nodes*:

Table 3.35: NO *Nodes* data density summary

Attribute name	Description	Units	Data density [%]
exact	value indicating the accuracy in geo-referencing		100
elevation_m	elevation of the node	m	100

3.9.3 Copyright for the Norway data set

Data availability and data usage

The data is provided through the Norwegian Petroleum Directorate (NPD). Here, we only use their shapefile that contained the geo-referenced gas *PipeLines* element. However, further information regarding the Norwegian data can be found under:

https://www.norskpetroleum.no/en/production-and-exports/the-oil-and-gas-pipeline-system/

Copyright

The copyright regulations of this data can be found under (https://data.norge.no/nlod/en/) and is given as:

> "The licensee, subject to the limitations that follow from this licence, may use the information for any purpose and in all contexts, by:
>
> - copying the information and distributing the information to others,
> - modifying the information and/or combining the information with other information, and
> - copying and distributing such changed or combined information.
>
> This is a non-exclusive, free, perpetual and worldwide licence. The information may be used in any medium and format known today and/or which will become known in the future. The Licensee shall not sub-license or transfer this licence. © Norwegian Petroleum Directorate."

However, if you use this data or any data set which incorporates this data, you are also obliged to cite the original authors of the NO data as follows:

Gassco AS, The Norwegian Ministry of Petroleum and Energy www.norskpetroleum.no.

Data disclaimer

As can be found under the NPD web page, the data disclaimer is given by the Norwegian Petroleum Directorate as the following:

> "Positional data accuracy is, unless otherwise stated, within approx. +/- 300 m. NPD is not responsible for accuracy on data reported by third parties. Content shall not be used for navigational purposes."

3.9.4 Summary Norway data

The data set was downloadable from the Norwegian "nationalgrid" operator. Tools have been created to load the Norway shapefiles and make them accessible throughout the SciGRID_gas project.

The Table 3.36 summarises the elements of the NO data set:

Table 3.36: NO component element summary

Component Name	Count
Nodes	53
PipeSegments	43

In addition, the map in Figure 3.30 visualizes the data for Norway.

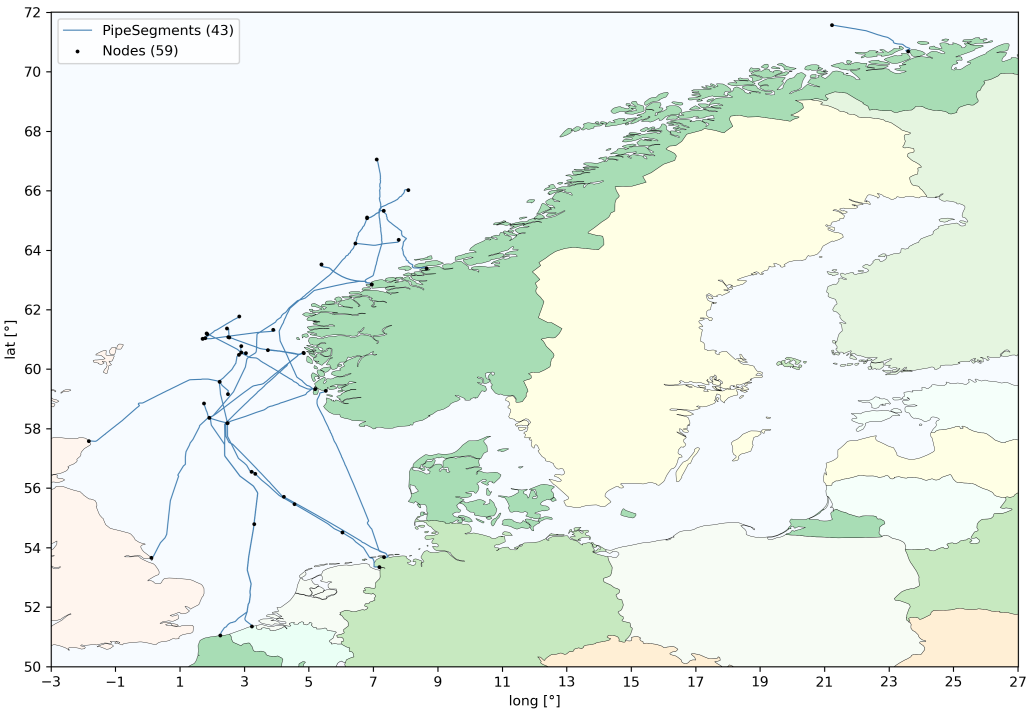


Figure 3.30: Overview of the NO data set.

3.10 Gas consumers data set

European gas consumers (**CONS**) is a further dataset for the SciGRID_gas project which was generated by extracting information from another non-OSM data source, and combining it with spatial information and distribution models. This work was carried out by Javier Sandoval as part of his Master's Thesis [San21]. The raw data generated is a time series of daily gas consumptions on a NUTS-3 level ([Wik21]) for all member states of the EU. A very detailed description of the data generation process can be found in [San21]. Here a brief summary of how the data was generated will be given, followed by its limitations and the extend of the data (data density).

3.10.1 Input data sets for the CONS data set

Several different data sets and information were used to generate the gas demand data, such as the spatial framework, gas consumed, number of household and more. Each of those input data sets will be listed and briefly described below.

Spatial disaggregation

Here for the spatial framework NUTS region were used, which is a geographical system dividing the European Union into regions of similar size in respect of similar number of inhabitants. Information associated with NUTS regions can be used for socio-economic analysing, policy analysis in respect of farming, and collecting, developing and harmonizing the European regional statistics. NUTS-0 is the coarsest division, describing the national level, meaning each country is a single NUTS-0 typology. NUTS-1 is the next coarsest division, with only 103 regions, NUTS-2 consists of 286 regions, whereas NUTS-3 consists of 1364 regions in the EU (including Norway which uses a similar classification and the former EU member UK, but excluding Malta and Cyprus). For the SciGRID_gas project the spatial information was given through shape files, such as:

- "NUTS_RG_01M_2016_4326_LEVL_1.shp"
- "NUTS_RG_01M_2016_4326_LEVL_2.shp"
- "NUTS_RG_01M_2016_4326_LEVL_3.shp".

In the SciGRID_gas project the gas consumption was modelled on the NUTS-3 level. For countries (e.g. Ukraine) which are not part of the EU and are not described by the NUTS-3, no disaggregation smaller than the national level could be carried out. In addition, further data sets were generated, using the CONS NUTS-2 and NUTS-1 were used. This allowed for a subsequent stronger simplification of the resulting gas transmission network data set.

Consumer type disaggregation

Next to the spatial disaggregation, a consumer type disaggregation was carried out. This information was available from the European Statistics department [Eur21b]. This data source supplied annual gas consumption on a national scale, broken up into the three sectors *Residential*, *Commercial* and *Industrial*. In addition, the European Statistical Office (EuroStat) also made available disaggregated final energy consumption of household and several industry consumers [Eur21c], which was a further input to the process. The three different sectors are defined as follow:

- *Residential*: Gas used by households due to space heating, cooking and hot water generation
- *Industrial*: Gas consumed by industrial facilities
- *Commercial, trade and services*: Gas consumed by smaller commercial, trade and service businesses, such as restaurants, workshops and other service providers (CTS).

Further independent variables

Previously it had been demonstrated by the DemandRegio project [GGB+20] that the gas demand depends on independent variables, such as temperature (household space heating uses gas in many countries), and GDP (industry production is related to gas consumption). Here a list of all further input data sets is given:

- Temperature: Average daily air temperature at 2 m above ground, for the years 2010 to 2019, on a spatial resolution of 0.25 degrees through ERA5 [CCCS21]. Here a method was implemented, that generated a single daily value for each NUTS-3 region. This single daily value was used in determining "heating degree days", which was the variable used.
- Number of households: Annual number of household by degree of urbanization on a NUTS-2 level [Eur21f], which was disaggregated onto a NUTS-3 level during the simulation process using population information.
- Population: Annual number of people living in NUTS-3 regions [Eur21g].
- Average dwelling area: Average size in units of square meters of dwelling by household type and degree of urbanization on a NUTS-0 level for the year 2012 only [Eur21a].
- GDP: Gross domestic product (GDP) at current market prices by NUTS-3 regions [Eur21e] on an annual basis for the years 2010 to 2019.
- Number of employees: Annual number of employees on a NUTS-2 level for the years 2010 to 2019 [Eur21d], which was disaggregated onto a NUTS-3 level during the simulation process using GDP information.
- Energy consumption: National annual gas consumption per the three sectors [Eur21b].
- Industry sector energy consumption: Information on different type of industry was available [Eur21b], which was linked to the number of employees by type of industry [Eur21h], resulting in larger gas consumption for regions with energy-intensive activities.
- National gas cooking: National values of cooking with gas [Eur21c] next to other energy sources
- National gas water heating: National values of water heating with gas [Eur21c] next to other energy sources
- National gas space heating: National values of space heating with gas [Eur21c] next to other energy sources.

Below for each sector and subsector the independent input variables are listed:

- *Residential*:
  - Cooking:
    - National gas cooking
    - Number of households.
  - Space Heating:
    - National gas space heating
    - Average dwelling area
    - Number of households
    - Temperature.
  - Water heating:
    - National gas water heating
    - Population.
- *Industrial*:
  - Energy Consumption: for sector industry

  - Number of employees in industry
  - Industry sector energy consumption
  - GDP.
- *CTS*:
  - Energy Consumption: for sector CTS
  - Number of employees in CTS
  - Temperature
  - GDP.

In addition, Sandoval incorporated a weekday dependent gas consumption for the sectors *Industry* and *CTS* (see [San21]).

With the above information Sandoval was able to estimate a gas consumption for all three main sectors on daily time steps for the years 2010 to 2019 (incl.). In addition a similar data set was available for Germany through the DemandRegio project ([GGB+20]) for the year 2015. Sandoval was able to calibrate his results with the DemandRegio gas consumption and derive fitting parameters, which were applied to all European countries, resulting in a daily time series of gas consumption for all NUTS-3 regions. The resulting gas demand time series can be accessed through the SciGRID_gas project.

3.10.2 Incorporating the CONS data into the SciGRID_gas data frame

This above described gas demand data set will be part of the data repository on Zenodo. In addition, Sandoval generated gas demand time series for NUTS-2 and NUTS-1 level as well, which will also reside in the same repository. If any of these data sets need to be incorporated into the SciGRID_gas project, then one needs to copy the data, in respect of the NUTS-levels, into one of the following folder: "Eingabe/Consumers/Nuts_3", "Eingabe/Consumers/Nuts_2" and "Eingabe/Consumers/Nuts_1".

However, due to the volume of the gas data demand sets size (NUTS-3 is about 140 MB of data), the time series data was reduced to the following attribute values and will be incorporated into the SciGRID_gas network data set as described below:

- *min_demand_M_m3_per_d*: The minimal value of gas consumed of the 10-year time series of the combined sectors (sum) data set.
- *max_demand_M_m3_per_d*: The maximal value of gas consumed of the 10-year time series of the combined sectors (sum) data set.
- *mean_demand_M_m3_per_d*: The mean value of gas consumed of the 10-year time series of the combined sectors (sum) data set.
- *median_demand_M_m3_per_d*: The median value of gas consumed of the 10-year time series the combined sectors (sum) data set.
- *min_demand_household_M_m3_per_d*: The minimal value of gas consumed of the 10-year time series of the sector household only.
- *max_demand_household_M_m3_per_d*: The maximal value of gas consumed of the 10-year time series of the sector household only.
- *mean_demand_household_M_m3_per_d*: The mean value of gas consumed of the 10-year time series of the sector household only.
- *median_demand_household_M_m3_per_d*: The median value of gas consumed of the 10-year time series of the sector household only.

- *min_demand_industrial_M_m3_per_d*: The minimal value of gas consumed of the 10-year time series of the sector industry only.
- *max_demand_industrial_M_m3_per_d*: The maximal value of gas consumed of the 10-year time series of the sector industry only.
- *mean_demand_industrial_M_m3_per_d*: The mean value of gas consumed of the 10-year time series of the sector industry only.
- *median_demand_industrial_M_m3_per_d*: The median value of gas consumed of the 10-year time series of the sector industry only.
- *min_demand_commercial_M_m3_per_d*: The minimal value of gas consumed of the 10-year time series of the sector CTS only.
- *max_demand_commercial_M_m3_per_d*: The maximal value of gas consumed of the 10-year time series of the sector CTS only.
- *mean_demand_commercial_M_m3_per_d*: The mean value of gas consumed of the 10-year time series of the sector CTS only.
- *median_demand_commercial_M_m3_per_d*: The median value of gas consumed of the 10-year time series of the sector CTS only.

Consumers elements

As described in the above subsection, a number of attribute values have been generated for each NUTS region. In addition, only the following two attributes are part of the "Consumers" component:

- nuts_id: Where the attribute name will be followed by an underscore and a number indicating the nuts level, e.g. "nuts_id_3" indicating that this is a NUTS-3 level region.
- exact: Accuracy in respect of geo-referencing (see Chapter 10.3).

This data set contains NUTS-3 regions of the European Union and other non-EU countries. Hence for those 1506 elements of type *Consumers*, consisted of 1348 values for NUTS-3 regions, and another 158 non-NUTS regions, e.g. locations in Albania, Switzerland and Turkey. For countries such as Ukraine, Belarus, Moldavia and Russia no NUTS values are defined.

Nodes elements

Overall, there are 1506 *Nodes* elements in the CONS data set. Each node contains information in respect of the standard attributes, such as "name" and "node_id", next to "elevation_m" and the attribute "exact" describing the accuracy in respect of the geo-referencing. As this info was available for all nodes, the data density is 100% for these attributes.

3.10.3 Copyright and disclaimer for the INET data set

Copyright

A list of the sources used for the generation of the CONS data set can be found in [San21].

Disclaimer

The CONS data set is supplied on a best-effort basis only, using available information as documented gathered through the process described in [San21]. While every effort is made to make sure the information is accurate and up-to-date, we do not accept any liability for any direct, indirect, or consequential loss or damage of any nature–however caused–which may be sustained as a result of reliance upon such information.

3.10.4 Summary CONS data

The CONS data set supplies information on gas consumption on a NUTS-3 level for Europe. This gas demand was generated through an interplay of independent variable information such as GDP, temperature and others, in combination with heuristic models, resulting in a daily time series spanning the years 2010 to 2019. This time series information was reduced to single values of minimum, median, mean and maximum gas demand for each NUTS-3 region for the three different consumer sectors: household, industry and commercial. The time series data and the summary data are accessible through the Zenodo web page, where the latter has been incorporated into a SciGRID_gas data set.

Below a table summarises the number of elements for each component:

Table 3.37: CONS component summary

Component Name	Count
BorderPoints	0
Compressors	0
Consumers	1506
LNGs	0
Nodes	1506
PipeSegments	0
Productions	0
Storages	0

In addition, the map in Figure 3.31 visualizes the data for Europe.

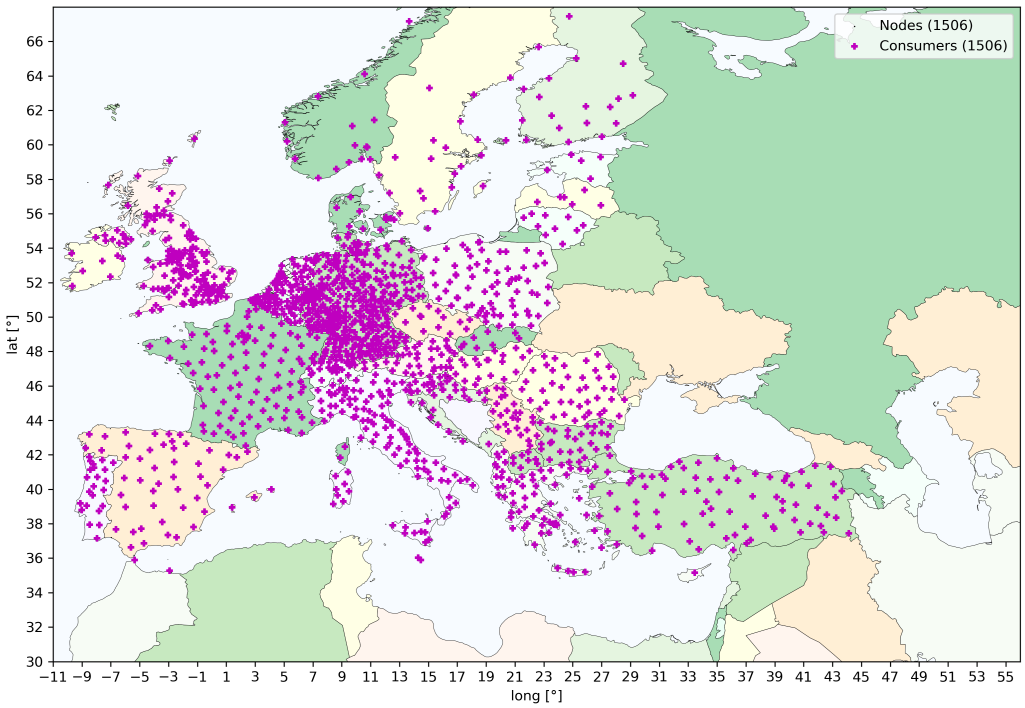


Figure 3.31: Overview map of the CONS data set for Europe.

3.11 Data summary

SciGRID_gas is based on open source data. To generate a gas pipeline network data set, one needs to access different data sets that were found throughout the project and presented here. Emphasis was given to depict the number of elements per component and the data density for each data set.

CHAPTER

FOUR

MERGING DATA SOURCES

Several gas facility data sources have been described in Chapter 3. Part of the SciGRID_gas project is to join (merge) those sets. However, some facilities might be present in more than one data source containing same but also different attributes. Hence, this chapter here will describe the current implemented methods to merge elements from different data sources.

4.1 Merging single node elements

So far, only individual data sources have been read in and converted to SciGRID_gas data sets. However, some elements might be described in more than one data set. In addition, they might be populated with different attributes and attribute values. Hence, those elements need to be merged, in such a way that the topology stays correct and that the attributes are merged correctly, while maintaining maximum information. The tools developed here have been designed for the non-OSM data sets. However, they should be applicable to any gas component data set. In addition, the concept and tools presented in this subsection apply to single node elements only. With this it is meant that elements of type *PipeLines* and *PipeSegments* will not be covered in this subsection, as they are elements containing more than one node. Merging tools for *PipeLines* and *PipeSegments* will be presented in subsection Chapter 4.2.

In this section here, the problem of duplicate elements from different data sets is being described with the help of some mock data set. This is used to describe the methods that have been implemented. Applying those tools will result in a single data set, not containing any duplicate non-pipe elements.

4.1.1 Problem description

In the Figure 4.1 the problem is depicted. There are elements from different data sets (different colour) with different attributes. For this example (summaries in Table 4.1) three different data sets are depicted for three different *Storages* elements: blue, red, and yellow. In addition, the spatial separation has been supplied in km.

Table 4.1: Summary of data of the three sample *Storages* elements.

Attribute name	Blue data set	Red data set	Yellow data set
name	Atwick	Aldbrough1	Aldbrough
max_cap_pipe2store_M_m3_per_d	1.9	1.0	1.1
max_cap_store2pipe_M_m3_per_d	2.3	1.3	
max_workingGas_M_m3		1800	
store_type	Depleted Field		Salt cavern

As can be seen, some elements have attribute values for the same attribute, and others do not, and the main question is: **Which elements should be merged, and which ones should not be merged?** Should all three be merged, because they are so close to each other, or none, as they all have different names, or just the red and the yellow one?

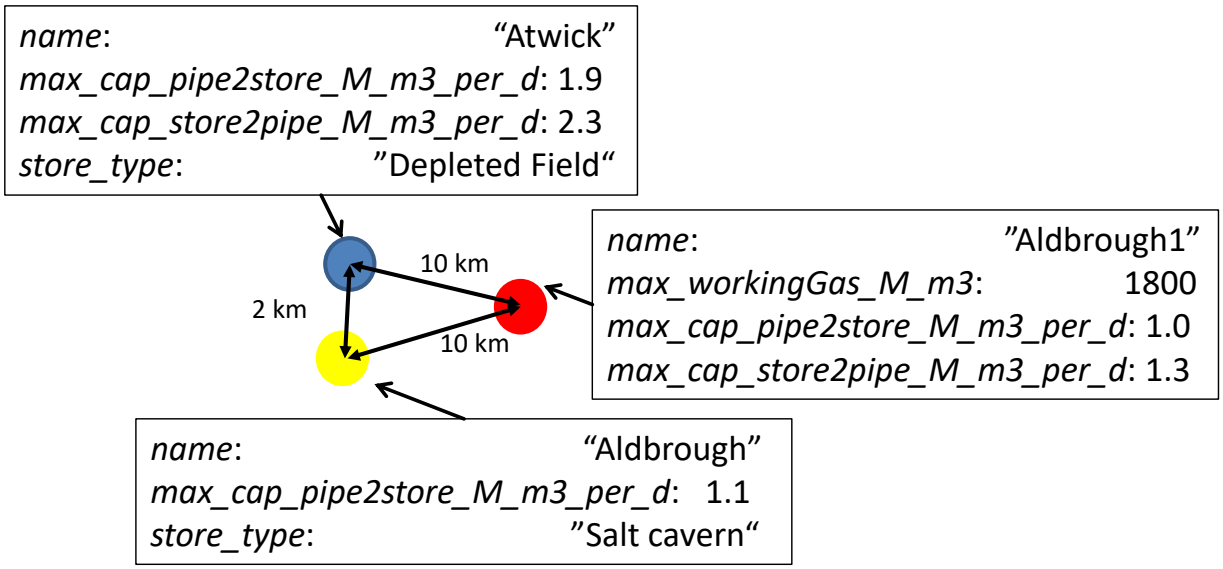


Figure 4.1: Example data sets blue, red and yellow, all depicting a storage element with different attributes and attribute values. The figure also includes the spatial separation between the elements.

Here approaches including name similarity, topological distance and country location have been developed so that an automated process can merge those elements that should be merged, and will be described next.

4.1.2 Methods for element identity comparison

Below different methods are introduced briefly to determine, if facilities from different data sets should be merged or not. The functions introduced are applicable to all components except *PipeLines* and *PipeSegments*. Different functions look at different attribute values, such as *names*, *LatLong* or *country code*. Each function returns a score between 0 and 100, with 0 indicating that there was no match between the attribute values supplied, whereas a score of 100 refers to a perfect match of the attribute values. In a second step, the user can set threshold values, so that a function returning a score larger than the set threshold is assumed to be describing the same facility. However, as single attribute comparison might be misleading (e.g. very similar place name in two different countries), therefore multiple attribute comparison methods have been introduced.

Spatial distance

This method determines if two elements can be classed as the same by their geo-reference location, by calculating the distance between the two locations. If the distance between two elements is 500 km, then it would be highly unlikely that the elements are of the same facility. However, if the distance between the locations is 10 km or less, then in the scheme of Europe they might be describing the same facility. The distance can be weighted in different ways, such as just the inverse distance, the inverse power distance or the inverse **log** distance. Different methods work best for different components and data sets. Here again, if the lat/long of two elements is equal, then this function would return a score of 100, whereas if the distance between two elements is several 100 or even 1000 km, then this function would return a very small score, such as 10 or 5, depending on the method selected.

Name comparison

This method helps to determine if two elements describe the same facility by comparing their location *names*. For the method return score of 100, the location names are identical, whereas for a method return score of 0 there is no similarity between the names at all. The names of "UGS Stollen" and "Stollen" would return a score of about 70, as the word "Stollen" is partially in the name "UGS Stollen". The external python library implemented here, returning score values for pairs of names, is called "FuzzyWuzzy".

An additional aspect was added to this method. If one entire name is part of the other name (**name-in-name**), then the user can specify that the return score should be increased by a user specified value. For the case where the names are "Aldbrough" and "Aldbrough1", and the name-in-name score be 100, then the final score would be 195. However, if the name-in-name score was not implemented, then the overall output score for the location name pair of "UGS Stollen" and "UGS" would be 43 only. In case that the user had specified a threshold score of 60, the storages with names "UGS Stollen" and "UGS" would not be merged, if the additional name-in-name would not have been implemented. Better results were achieved with the additional name-in-name method.

Country code comparison

Another key factor in determining if two elements should be merged is their corresponding country-code. In case that the country-codes of the elements are known, one can determine the equality of the country-code.

This method only returns two values:

- "0": the country code entries of the two elements are different.
- "100": the country code entries of the two elements are the same.

Combining comparison methods

However, it was experienced that individual single method selections did not result in the expected element selection. From the above example, if one had selected the spatial distance method, then "Atwick" might be merged with "Aldbrough", as they are closer than "Aldbrough" and "Aldbrough1". Whereas comparing the names only would have merged "Aldbrough" and "Aldbrough1". So which method should be used?

Hence, combinations of the above methods were developed. This is achieved by executing the methods subsequently to each other, resulting in a combined method return score. Two elements were deemed to be the same, if the final score was larger than a user specified threshold score.

Example results

Above an example was given in Figure 4.1. "Atwick" and "Aldbrough" are only 2 km apart, whereas "Aldbrough1" is separated by 10 km to any of the other two elements.

First of all, the method return score was determined for the spatial separation of the elements. The "Atwick"-"Aldbrough" spatial separation led to a method return score of 50, whereas the same value for the "Aldbrough" and "Aldbrough1" pair was 10 only. In a second step the names were compared, resulting in method return scores of 0 and 195 respectively. Hence, for a user specified threshold score of 60, only the elements "Aldbrough" and "Aldbrough1" would be merged. As can be seen in Table 4.1, the elements "Aldbrough" and "Aldbrough1" have same and complementing attributes. The attributes that the resulting merged element will have will be described in the subsection below.

Attributes of resulting element

As can be seen in Table 4.1, "Aldbrough" and "Aldbrough1" contain a mix of different and same attributes, with partially different values for the same attribute. Here the following attribute merge path is being implemented:

- Assume that "Aldbrough1" will be merged into "Aldbrough".
- The resulting element will have all those values from element "Aldbrough"
- Any element that was not given through "Aldbrough", and is present in "Aldbrough1" will be copied to "Aldbrough".

Hence, the resulting "Aldbrough" element would have the following attributes with the following values:

- *name*: "Aldbrough"
- *max_cap_pipe2store_M_m3_per_d*: 1.1
- *store_type*: "Salt cavern"
- *max_workingGas_M_m3*: 1800
- *max_cap_store2pipe_M_m3_per_d*: 1.3.

Summary

The above text example was used to explain the merge process of single node elements, such as *LNGs*, *Storages* and *Productions*. This section will be followed by explaining a method that can be used for merging pipes, which are elements connecting more than one node.

4.2 Merging pipe elements

Merging elements that are associated with a single node (point) has been described in subsection Chapter 4.1. However, a more complex method is needed to be implemented to merge *PipeSegments* or *PipeLines*. This process will be explained below.

In this section, the problem of duplicate elements from different data sets is being described with the help of some mock data sets. They will be used to describe the methods that have been implemented to achieve the merge task. This will try to achieve a data set, which will not contain any duplicate elements for the components *PipeSegments* and *PipeLines*.

4.2.1 Problem description

In the Figure 4.2 the problem that the SciGRID_gas project needs to solve is depicted. There are two individual network data sets (subplot a)): network "B" (in blue) and network "R" (in red). By looking at the example, one could argue that pipes B3 and R1 are describing the same real facility, and pipes B4 and R2 are also describing the same facility. Here it is already assumed that all other network pipes are only in one or the other network data set (as one can see, *PipeSegments* B1 and B2 are only in the blue network, whereas *PipeSegments* R3, R4 and R5 are only in the red network).

Further to the spatial visualization, Table 4.2 summarizes the attribute values for each element of the two data sets.

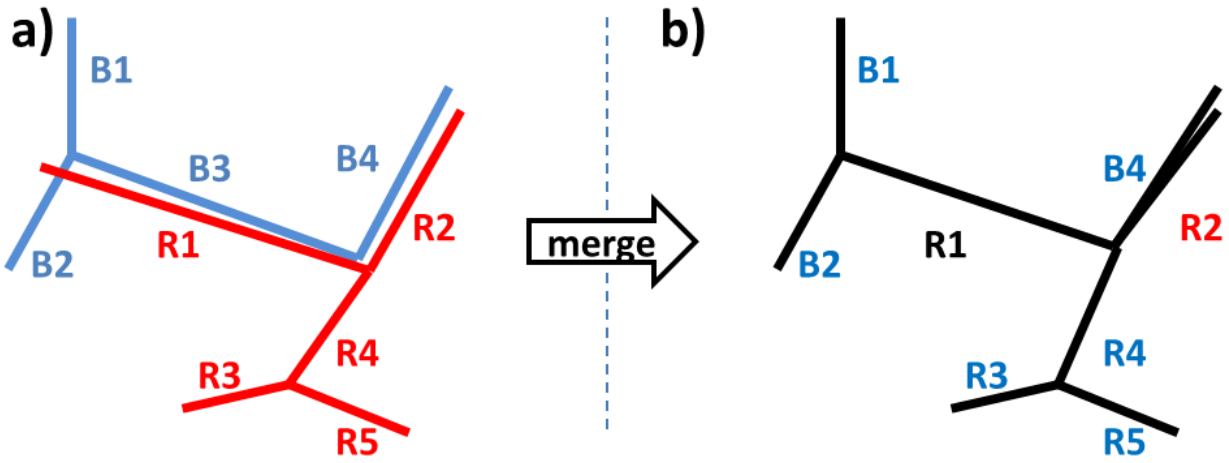


Figure 4.2: Example network data sets blue (B1..B4) and red (R1..R5). Figure also indicates the spatial separation between the nodes. Subplot a) depicts the individual networks, whereas subplot b) depicts the merged data set.

Table 4.2: Summary of key attribute values of the elements of the two networks "B" and "R".

Pipe	diameter_mm	max_pressure_bar	max_cap_M_m3_per_d	length_km
B1				5
B2	900	85		4
B3	900	90		10
R1	910			10.9
B4	700	70	10	5.5
R2	900	90		5.6
R4	1100			3.8
R3	600			2
R5	800			3

However, the main question is: **Which pipes should be merged, and which ones should not be merged?**

Here, methods of name similarity, topological distance, country location and key attribute similarity have been developed as part of the SciGRID_gas project and will be described next. This will allow an automated process to merge those elements that should be merged, and not merge those pipes that are too different, e.g. through attribute values, which will be explained below.

4.2.2 Methods of identifying identical pipelines

Below different methods are being introduced that are being used to help in determining, if pipes from different data sets should be merged or not. Some of the functionality used has been introduced already in Chapter 4.1. Other additional functions are being introduced below.

The overall pathway of determining if two pipelines from two different data sets are the same is presented in Figure 4.3, and consists of the following steps:

- Determine if start and end nodes from different data set are the same
- Determine if the attribute values diameter are similar
- Determine if the attribute values maximum operating pressure are similar

- Determine if the attribute values maximum flow capacity are similar
- Determine if the attribute values pipe length are similar.

If the process determines yes for all those requirements, then it is assumed that the two pipes are the same and should be merged. Each of those processes will be described in more detail below.

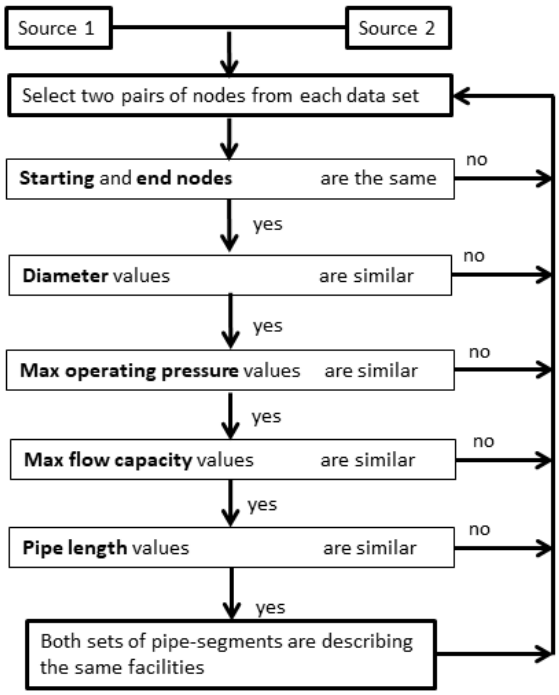


Figure 4.3: Process flow chart for determining, if two pipes from two data sets networks are describing the same pipe.

Testing: End node location similarity

This process test, if the geolocation of two nodes from two different datasets are similar, as has been described in Chapter 4.1. If those nodes are describing the same location, then the pipes connecting those nodes can be determined and the next steps can be carried out. Otherwise, another combination of nodes will be tested.

Testing: Pipe diameter similarity

After identifying two pipelines with the same start and end nodes from two different data sets, the pipeline diameters are compared. As can be seen from the example listed in Table 4.2, the pipes B3 and R1 do have similar values, 900 and 910 respectively. Here, the user specifies, by how much these values are allowed to be different. In the SciGRID_gas project, this difference (tolerance) is given in respect of percent, and a value of 5% has currently been implemented. Hence, B3 and R1 have similar diameter values, whereas pipe B4 and R2 would not fulfil this requirement.

Testing: Maximum operating pressures similarity

The next test is regarding the similarity of the maximum operating pressure. Currently a tolerance of 5 % has been implemented into the SciGRID_gas project. If two pipeline pressure values are within 5 % of each other, then the values are deemed to be the same.

However, with the example given above, incomplete information is being supplied, which will be the case for many merge attempts. Pipes B3 and R1 have the maximum operating pressure values of 90 and None, respectively (see Table 4.2). As only one pipe has a value, the following rule has been implemented: If one or both elements have no given attribute value, then the test is assumed passed. Therefore, for B3 and R1, this test returned true, and it is assumed that the pipes from those two networks describe the same pipe.

Testing: Maximum flow capacity similarity

The next test is comparing the maximum gas flow capacity. Currently a tolerance of 10% has been implemented into the SciGRID_gas project. If two values are within 10% of each other, then the values are deemed to be the same.

Based on the above example from Table 4.2, where pipes B3 and R1 have no values, the rule introduced for the previous attribute applies. Hence, one can assume that B3 and R1 both describe the same pipe.

Testing: Pipe length similarity

The next test is checking if the summed lengths of the individual pipes are similar or not. Different tolerance values have been implemented for different data sets. For the situation, where the INET and the GB data sets need to be merged, a tolerance of 30 % was set. The GB data set originates from topological correct pipes path ways supplied through a shape file, where each bend or corner of the pipes can be found in the shape file. The INET data set only connects nodes with straight lines, hence, ignoring the actual path. Hence, the variation in pipe length can be large between those two data sets.

A different tolerance value was applied to the Norway (NO) data set, when merging with the INET data set. The NO data set also originated from a shape file. However, the pipes are (due to missing obstacles on the sea bed) dominantly straight lines, like connection nodes via the shortest path, which is also the case for the INET data set, hence, the tolerance value was set to 10 %.

For our above example, where the pipe lengths were 10 and 10.9 km for the B3 and R1 pipes respectively, the variation in the attribute value is less than 10 %, and hence, those two pipes would also pass this test.

Results of sample merge

Hence, after applying the above method path way to the pipes from the two networks, in the resulting network the original pipes B3 and R1 were merged, whereas the pipes B4 and R2 were not merged (Figure 4.2 b)). Here the colour coding of the pipe labels indicates, from which data set the attributes of the resulting pipes originated from. Blue means that all attributes for this pipe come from the corresponding blue pipe. Red indicates that the attributes originate from the red network data set. And black indicates that the resulting pipe contains attributes from both data sets.

Comment

It should be pointed out that there might be multiple parallel pipelines in one or both data sets. Here it is assumed that parallel pipelines within the same data set are describing different pipelines, and hence, are not allowed to be merged. However, the method described above can be applied to a situation, were one or both data sets contain parallel pipes. E.g. if data set "R" has a set of parallel pipes (**R1** and **R2**) between nodes **A** and **B**, and the second data set "B" has three parallel lines between (**B1**, **B2**, and **B3**) between nodes **A'** and **B'**, then the above method would test if the following pipe combinations could be merged:

- 1) **R1** with **B1**
- 2) **R1** with **B2**
- 3) **R1** with **B3**
- 4) **R2** with **B1**
- 5) **R2** with **B2**
- 6) **R2** with **B3**.

If through the above process it is found that **R1** should be merged with **B2**, then the test for **R1** and **B3** would not be carried out any more, and would jump to testing **R2** and **B1** next. Hence, the system was designed in such a way that it can also deal with identical pipelines from the same data set (e.g. same length, same pathway, same pressure, and max capacity). This was achieved by removing merged pipelines from the pool of mergeable pipelines.

Merging INET with EMAP

The above described process was used to merge the EMAP pipes with the INET pipes, and this process worked well. It worked well, as the EMAP data set consisted of more lines, and each pipeline had a higher spatial definition (way points between start and end node). This made the merging process "relatively simple", not really resulting in same pipelines from different data sources ending up twice in the final data set. However, this process was attempted to be used for merging the LKD/GB/NO data sets with the EMAP data set. However, it was found that the above process did not work to the standards required (e.g. too many pipes which were the same in both data sets ending up as parallel lines in the final data set). Hence a slightly varied merge process will be described in the following section for the LKD and NO data sets. Due to copy right issues, the GB was different again, and will be described last.

Merging EMAP with LKD/NO

As described above, the merge process that was used to merge the INET with the EMAP data sets could not be used for merging any of the LKD, NO data set with the EMAP data set. Hence an alternative approach has been implemented. The major difference is: the EMAP pipes for the Germany and Norway were removed and filled with the LKD, and NO data pipes. However, care was taken so that attribute values from the EMAP data set was written into the LKD and NO data set before removal. The process will be described in more detail below for the merge of the LKD and EMAP data set as an example. The same process was also applied to the NO data set.

The process can be described by the following steps:

- Determine if a pipe of the EMAP data set is also present in the LKD data set.
- Determine if they have the same attribute values, and hence are describing the same real pipe.
- Copy the attribute values from the EMAP pipe into the LKD pipe.
- Repeat the above process till no more pairs of pipelines can be found.
- Create a list of pipes that cross the German border (German cross-border pipes).
- Remove all pipes from the EMAP data set that are in the country of Germany.

- Copy all LKD pipes into the EMAP data set.
- Assure that the new pipes in the EMAP data set are connected to the other pipes in the EMAP data set, using the German cross-border pipes.

With the above process, it is assured that as little attribute information as possible is being lost when removing the pipes from the original EMAP data set, as those attribute values were copied to the LKD data set. The same process was being carried out for the data set of Norway.

Again, it is believed that this is the best possible method, as the other merge process did lead to multiple entries of the same pipe in the resulting merged data set, and it assured in retaining as many attribute values as possible. It is believed that this method is required, as the pipes in the EMAP data set for Germany, when compared with the LKD data set were almost "identical", however different enough, so that matching pairs could not be found.

Merging EMAP with GB

Here a match of the pipes from the EMAP and the GB map was carried out as has been described in the previous sections. However, only the attribute values from the GB pipes were copied into the EMAP data set, and not the actual pipes. Those attribute values can be used for heuristic processes and be replaced as part of this process as well, so that no information from this data set is passed on to other users, and the copy right rules are not broken.

Summary

The above subsection examples demonstrated how pipes from different networks can be identified to be describing the same physical pipe, so that the information from such pipes can be merged, and the pipes are not duplicated in a merged network data set. For this a method pathway flow chart of the tests/methods has been introduced that pipe pairs need to fulfil to be considered identical pipes. With this, networks can be merged, so that duplicate pipeline elements have been removed. Due to different granularity of the data sets and copy right restrictions, different methods of merge processes were implemented.

4.3 Application to the INET, GIE, GSE, IGU, EMAP, LKD, GB, NO and CONS data sets

For the data sets of INET, GIE, GSE, IGU, EMAP, LKD, GB, NO and CONS, elements from the following components needed to be merged:

- *Compressors*
- *LNGs*
- *PipeSegments*
- *Productions*
- *Storages*
- *Nodes*.

In addition, the following components were given through a single raw data set:

- *BorderPoints*.
- *PowerPlants*
- *Consumers*

Each of those components will be described below.

4.3.1 Merging *BorderPoints*

Elements from the component *BorderPoints* are only present in the data set INET, hence, no merging of elements from different data sets is required. The final number of *BorderPoints* elements is 119.

4.3.2 Merging *PowerPlants*

Elements from the component *PowerPlants* are also only present in the data set INET, hence, no merging of elements from different data sets is required. The final number of *PowerPlants* elements is 331.

4.3.3 Merging *Consumers*

Elements from the component *Consumers* are also only present in the data set CONS, hence, no merging of elements from different data sets is required. The final number of *Consumers* elements is 1506, when the NUTS-3 was selected. For the NUTS-1 and NUTS-2 the number of *Consumers* is 120 and 323 respectively.

4.3.4 Merging *Compressors*

To be able to assure that compressor elements from different data sets should be merged, a combination of "name" and "spatial distance" was implemented. An overall threshold score of 20 was set for the data sets. Hence, in a first step the distance was investigated. Here the inverse method was selected. For this method the spatial distance between two elements is being determined and returned as distance in units of km. Then the following equation was carried out: $score = min(100/distance\_km, 100)$.

In a second step the method return score for the name is being determined. Values can range between 0 and 200, as the name-in-name method (see Chapter 4.1) was also included.

In the final step both method return scores were added. And element pairs with a value of 20 or larger were deemed to be the same and were merged.

For the *Compressors* terminals only the INET, the LKD and the GB data sets contained any information on the component *Compressors*. It was determined that the user specified threshold value of 20 works best for *Compressors* elements.

The number of elements per data set are listed in Table 4.3. The table shows that the merged data set has the same number of elements as the INET input data set. This indicates that the LKD data set and the GB data set did not contain any new facilities that were not present in the INET data set. However, possible better geo-referencing of the elements from the LKD or GB data sets can led to better geo-reference values in the final data set.

Table 4.3: Number of *Compressors* elements per input data set and merged data set.

Data set	Number of *Compressors* elements
INET	249
GIE	0
GSE	0
IGU	0
NO	0
LKD	13
GB	21
EMAP	0
CONS	0
Merged data set	249

4.3.5 Merging *LNGs*

Merging *LNGs* terminals follows the same path as described for merging *Compressors* elements. Here only the INET and the GIE data sets contained any information on the component *LNGs*. It was determined that the user specified threshold value of 25 works best for *LNGs* elements as well.

The number of elements per data set are listed in Table 4.4. The table also indicates that the merged data set has the same number of elements as the INET input data set. This indicates that the GIE data set did not contain any new facilities that were not present in the INET data set. However, additional attributes and attribute values were supplied through the GIE data set, resulting in "better" data of the merged data set.

Table 4.4: Number of *LNGs* elements per input data set and merged data set.

Data set	Number of *LNGs* elements
INET	32
GIE	21
GSE	0
IGU	0
NO	0
LKD	0
GB	0
EMAP	0
CONS	0
Merged data set	32

4.3.6 Merging *PipeSegments*

Merging *PipeSegments* follows the path as described in Chapter 4.2. Here the INET, the NO, the LKD, the GB and the EMAP data sets contained information on the component *PipeSegments*.

The number of elements per data set are listed in Table 4.5. The table indicates that the merged data set increased significantly in numbers of pipes.

As described in Chapter 3.9, Chapter 3.7, Chapter 3.8 and Chapter 3.6 the data sets of NO, LKD, GB and EMAP are of slightly different nature. The NO data originates from a shape file with very high topological accuracy. The LKD data set originates from a shape file, however, with lower topological accuracy. The GB data set originated from a shape file with very high topological accuracy as was tested manually. Further, the EMAP data set was generated by converting an topological PDF map into an electronic version, resulting in lower topological accuracy. Hence, for merging any of those data sets with the INET data set, different merging function threshold values were implemented. For the NO data set, a threshold of 30 was selected, the LKD data set had a threshold value of 2.5, for the GB data set a threshold value of 5 was set, and for the EMAP data set a function threshold value of 20 has been set. All of those threshold values were estimated using testing processes.

Table 4.5: Number of *PipeSegments* elements per input data set and merged data set.

Data set	Number of *PipeSegments* elements	Length [km]
INET	920	60,206
GIE	0	
GSE	0	
IGU	0	
NO	43	9,321
LKD	1261	25,511
GB	229	7,661
EMAP	5146	207,622
CONS	0	
Merged data set	8461	291,590

4.3.7 Merging *Productions*

Elements from the component *Productions* are given through the LKD and EMAP data sets, where the EMAP data set supplied only a rough topological location, whereas the LKD data set supplied additional attribute values. To merge the two data sets a threshold value of 50 was used.

Table 4.6: Number of *Productions* elements per input data set and merged data set.

Data set	Number of *Productions* elements
INET	0
GIE	0
GSE	0
IGU	0
NO	0
LKD	6
GB	0
EMAP	117
CONS	0
Merged data set	109

4.3.8 Merging *Storages*

Merging *Storages* facilities follows the path as described for the component *Compressors*. Here the INET, the GIE, the GSE, the IGU, the LKD and the EMAP data sets contained information on the component *Storages*. It was determined that the user specified threshold value of 24.05 works best for *Storages* elements for all data sets, except for the EMAP data set were a threshold value of 50 worked best.

The Table 4.7 shows the number of *Storages* of the individual data sets prior to the merge process, and the resulting number of elements after the merge process. As can be see, by combining the GIE, GSE, IGU, LKD, GB and the EMAP data sets to the INET data set, 112 additional elements were added to the INET data set. In addition, part of the merge process was also the migration of the attributes and attribute values, which will be discussed in more detail in Chapter 7.

Table 4.7: Number of *Storages* elements per input data set and merged data set.

Data set	Number of *Storages* elements
INET	199
GIE	109
GSE	210
IGU	144
NO	0
LKD	14
GB	0
EMAP	238
CONS	0
Merged data set	311

4.3.9 Merging *Nodes*

Elements of type *Nodes* were not merged through those processes described for the other point elements. However, all the other processes above were leading to nodes being moved and merged. For completeness, in Table 4.8 the number of nodes for each original data sets has been listed. The final number of nodes after the merge process of the resulting merged data set is 7971.

Table 4.8: Number of *Nodes* elements per input data set and merged data set.

Data set	Number of *Nodes* elements
INET	1424
GIE	115
GSE	168
IGU	137
NO	53
LKD	938
GB	305
EMAP	4323
CONS	1506
Merged data set	4337

4.3.10 Summary

In Chapter 4 different merge selection methods were introduced. These were applied to the INET, GIE, GSE, IGU, NO, LKD, GB, CONS and the EMAP data sets for the components - *Compressors*, *LNGs*, *PipeSegments*, *Productions*, *Consumers*, *PowerPlants* and *Storages*.

4.4 Summary

When several data sets are combined, the situation can occur that the same facility is presented by two or more data sets. Instead of this facility being present several times in the final merged data set, methods were presented that would determine the likelihood that elements from different data sources are describing the same facility. Those element pairs were detected through a comparison of names, location and country-code values. Elements that were deemed to be the same were merged accordingly, so that individual facilities were only present once in the final data set. This chapter described in detail the pathway and method that has been implemented in the SciGRID_gas project to achieve this goal. The following section will describe the pathway and method how missing attribute values can be generated.

CHAPTER

FIVE

HEURISTIC ATTRIBUTE VALUE GENERATION

Gas facility data sources have been described in Chapter 3. However, those data sources might not contain values for all attributes, and hence, those values need to be generated. This chapter here will describe the current implemented heuristic methods that can be used to estimate missing attribute values. The can be grouped into two categories, the "physical-based" heuristic processes and the "statistical" processes.

With "physical-based" heuristic processes it is meant that some "physical" relation between attributes is know, using one (or more than one) independent attribute and a linear or non-linear relationship. E.g. as was described by Kunz et.al [KKS+17], the capacity of a pipeline can be related to the diameter, pressure and pipeline class of the pipeline itself. Hence in case of missing capacity values, and the presence of the diameter, pressure and pipeline class, this relationship should be used.

However, for alot of attributes, a "physical-based" heuristic relationship is not known, and one needs to apply the "statistical" process. Here, the independent attribute(s) can be related to the dependent attribute (missing attribute), and even though from a physical point of view, the relationship might not be realistig, however it does determine the missing attribute values, including an uncertainty.

In this chapter, different "physical-based" heuristic methods will be indroduced first. This will be followed by the introductin of the "statistical" heuristic processes.

5.1 Physical-based heuristic processes

5.1.1 Flow direction estimation

A further attribute that needs attention is the direction of the gas flow in the pipelines. In general, the data set has been defined in such a way that the gas flows from the first to the last node of each pipeline. However, for a large number of pipes, it is not known, if this assumption is correct. In addition, some pipes can be operated bi-directional. Hence a heuristic method has been implemented that determines the gas flow direction.

The overall thought behind the implemented approach is: "In a gas network the gas flows from the sources to the sinks."

In the network that has been created, there are several sinks and sources. The sinks are:

- *Consumers*
- *PowerPlants*
- *Storages*,

whereas the sources are:

- *LNGs*
- *BorderPoints*

- *Productions*
- *Storages*.

As can be seen, the component *Storages* appears under sink and sources, as storages get filled in the summer and their gas is being used in the winter. Hence any estimation needs to incorporate this aspect.

Overall, two different heuristic methods have been implemented. The first one ("NumPath") counts how often a pipe is being used to carry gas from a source to a sink, and pays attention to the direction of the gas flow. The second approach ("Capacity") stores the information of how much assumed gas flows from the sources to the sinks, also paying attention to the direction. Both approaches will be described here, whereas the "Capacity" has been implemented here for the SciGRID_gas project, but in case of unknown gas capacity values for the sinks and sources, the "NumPath" approach could be selected by the user.

Flow direction determination using "NumPath" method

Figure 5.1 depicts a schematic diagram for two consumers (sinks) and one LNG terminal (source). As can be seen, the flow of gas goes from the LNG terminal to the consumers, where the pipe from the LNG terminal to the junction is being used twice (for two downstream consumers), whereas the pipe from the centre junction to each consumer is being used only once. One needs to do that for all sinks, where the corresponding source is selected based on the shortest path length.

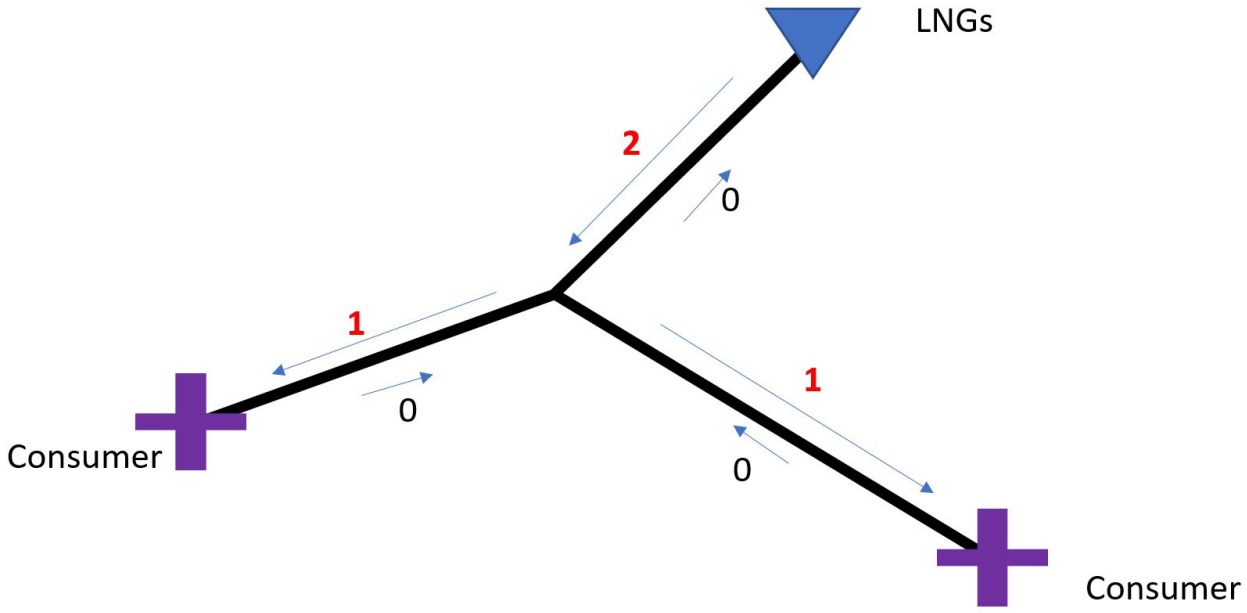


Figure 5.1: Schematic diagram of two consumers (sinks) and one LNG terminal (source).

As mentioned above, the elements from the component *Storages* are being treated as a sink in summer, whereas they are treated as sources in the winter, as can be seen in Figure 5.2. Hence one needs to combine the summer and the winter results, which is carried out by adding the flow counts of winter and summer for each pipe. As one can see, the pipe from the centre junction to the *Storages* element has gas flowing in both directions, whereas all other gas pipelines appear to have flow going in only one direction.

The following assumptions have been applied, trying to determine the directionality of the pipe, **and** if the pipe should be operated bi-directional. For this a *est_uniDirection_perc* attribute has been defined as depicted in Figure 5.3.

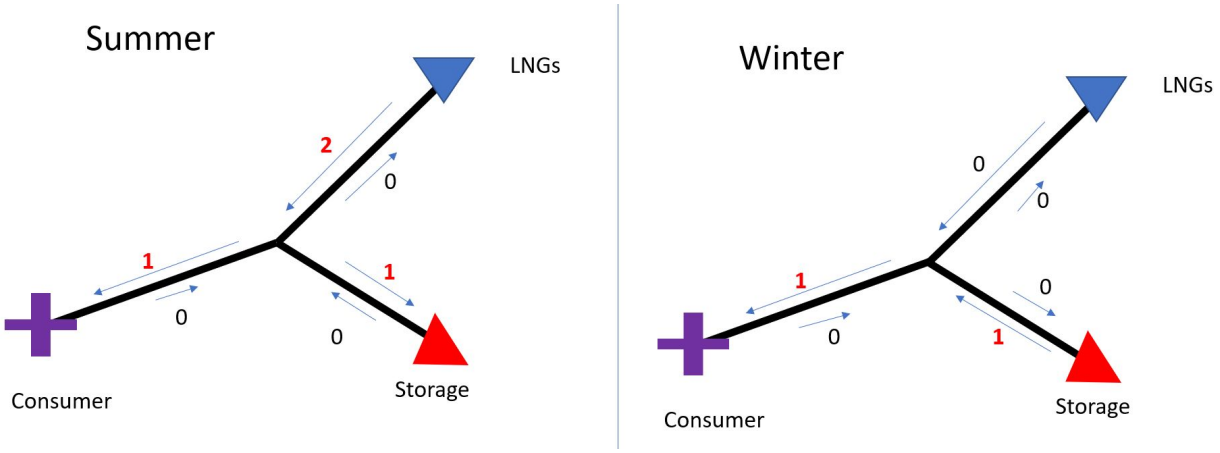


Figure 5.2: Schematic diagram of the flow of gas in summer (left) and winter (right).

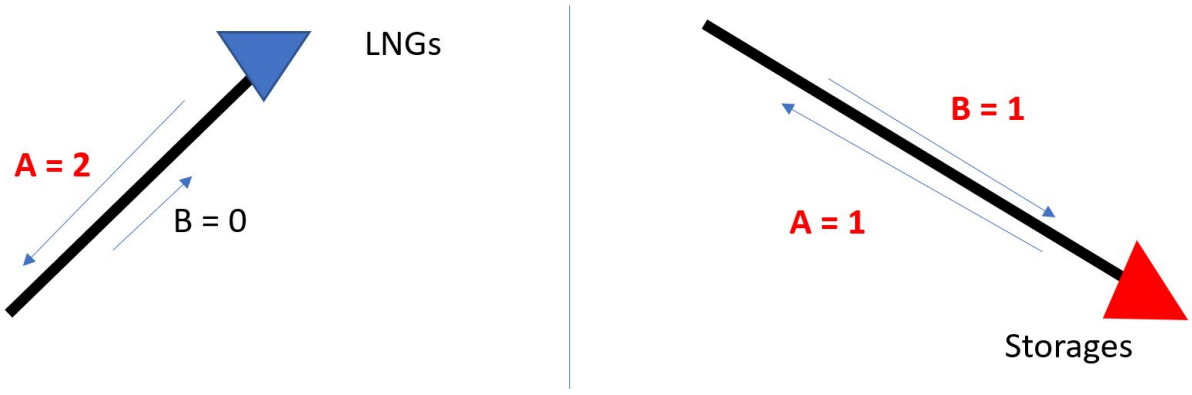


Figure 5.3: Schematic diagram of the flow of gas in summer (left) and winter (right).

To determine if a pipeline is bi-directional or uni-directional, the *est_uniDirection_perc* is defined as following:

$$est\_uniDirection\_perc = \frac{A}{A+B},$$

whereas *A* counts the gas flow occurrences from the start node to the end node, whereas *B* counts the gas flow occurrences from the end node to the start node.

For the above example (Figure 5.3, left) the value for *A* is 2, whereas the number of flows in the opposite direction is zero (*B* = 0). This results in a *est_uniDirection_perc* of 100. The following assumptions have been made for the *est_uniDirection_perc*:

- *est_uniDirection_perc* > 66.67: It is assumed that the pipe is oriented correctly (first node is start node, last node is end node), and that the gas is flowing predominantly from the start node to the end node, and hence is uni-directional.
- *est_uniDirection_perc* < 33.33: It is assumed that the pipe is oriented incorrectly (flow goes from the end node to the start node), and that the gas is flowing predominantly in one direction only, resulting in a uni-directional pipe.
- 33.33 < *est_uniDirection_perc* < 66.667: Here it is assumed that the gas is flowing in both directions. Hence the pipeline is deemed to be bi-directional. For a bi-directional pipeline, it does not matter, which node is the start and which node is the end node of the pipeline.

With the above definitions, the pipe on the left in Figure 5.3 is a correctly oriented uni-directional pipe, whereas the pipe on the right in Figure 5.3 is a bi-directional pipe. Results for Spain are depicted in Figure 5.4.

However, as can be seen, some problems arise with this method. E.g. for the pipe leading to the storage (red triangle) in the Pyrenees, a uni-directional pipe has been estimated, same for the storage close to Sevilla.

In addition, for the above method, one does not know if the pipelines are able to cater for the capacity required by the sinks. Hence, a "capacity" method has been implemented.

Flow direction determination using the "Capacity" method

For the "Capacity" method, the known supply and demand values of the individual elements are being used. E.g. one knows the maximum amount of gas that individual LNG terminals are able to pump into the network (*max_cap_store2pipe_M_m3_per_d*), or the maximum amount of gas that a consumer uses (*max_demand_M_m3_per_d*).

The initial approach for the "Capacity" method is the same as for the "NumPath" method, where the heuristic process determines which sink can be supplied from which source (selection through shortest path). The required sink capacity is added to the pipelines along the path, same way the number of connections were added to the pipeline in the "NumPath" method. However, the "Capacity" method also assures the following two aspects:

- Is there enough gas at the source to supply the demand?
- Do the pipes along the path between source and sink have enough capacities, to carry that additional gas from the source to the sink?

If in this process any of the two questions are answered with "no", then the heuristic process selects either a different source or a different pipelines between sink and source. The example of a different source is depicted in Figure 5.5.

Here the consumer (bottom left) requesting 3 $\mathrm{Mm^3 d^{-1}}$ can get the gas amount from the LNG terminal that can supply up to 10 $\mathrm{Mm^3 d^{-1}}$ (top middle), leaving another 7 $\mathrm{Mm^3 d^{-1}}$ for other consumers. For the consumer at the bottom right, which needs 5 $\mathrm{Mm^3 d^{-1}}$, the closest source would be the LNG terminal that can supply 4 $\mathrm{Mm^3 d^{-1}}$. However, not all of the demand for the consumer can be supplied by the LNG terminal. Hence a different source is selected through the heuristic process, in this case the LNG terminal top middle, which still has a capacity of 7 $\mathrm{Mm^3 d^{-1}}$.

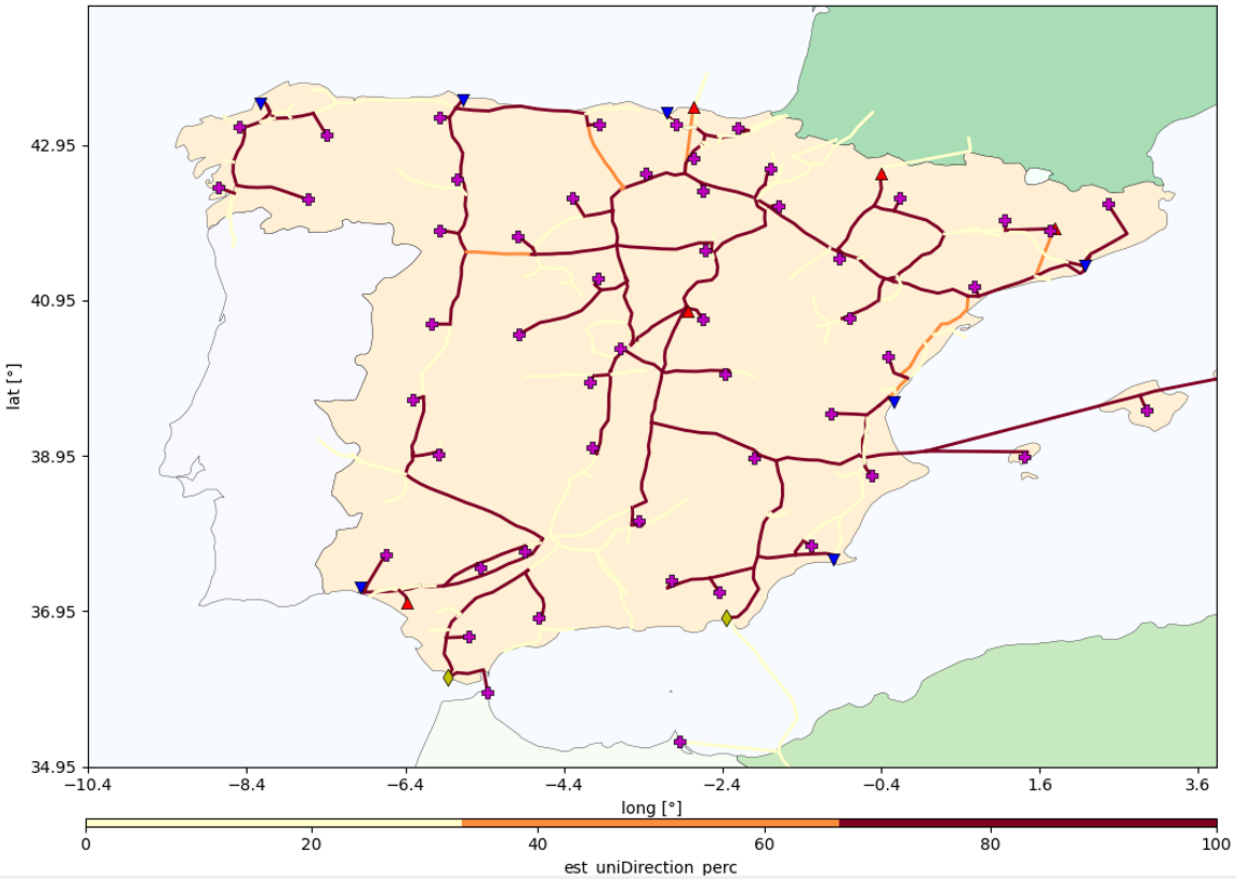


Figure 5.4: Estimated gas flow direction values for Spain, depicting *LNGs* (blue triangles), *Storages* (red triangles), *BorderPoints* (yellow diamantes), *Consumers* (violet crosses), and *PowerPlants* (green X).

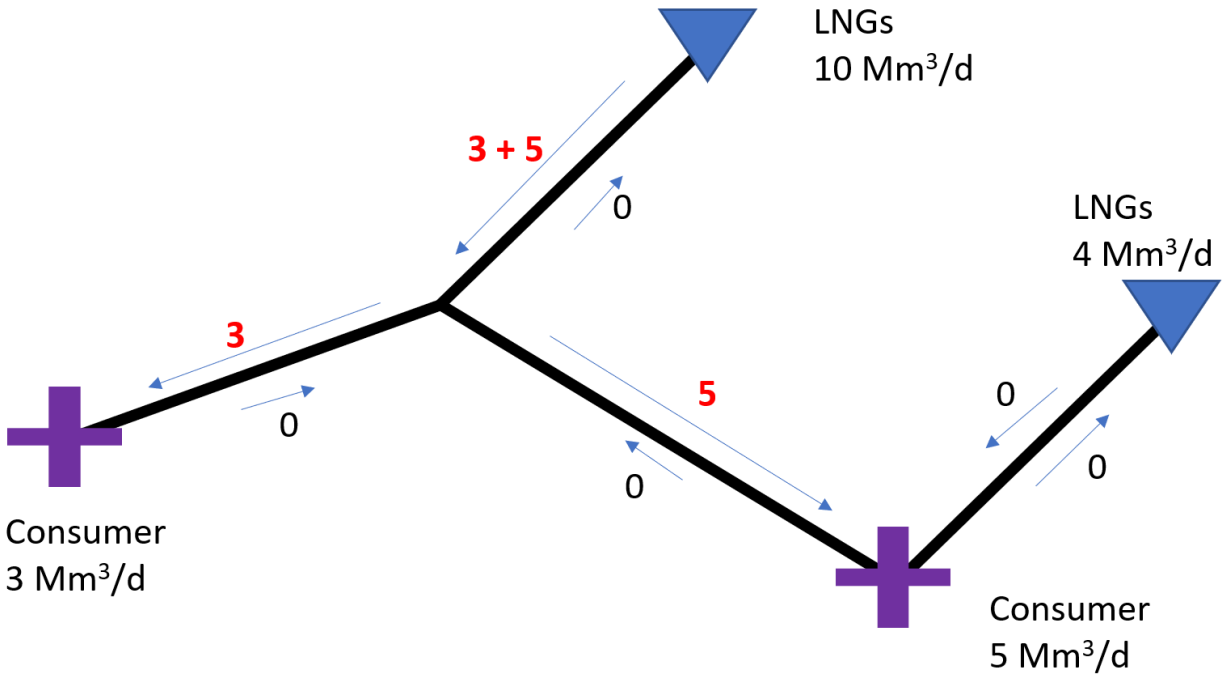


Figure 5.5: Schematic of pipe selection due to supply limitations.

The subsequent steps are similar to the steps undertaken for the "NumPath" method, where the heuristic process determines a value of *est_uniDirection_perc* for each pipeline. Directionality of the pipes is determined in the same way, however using the capacity values. The results for Spain can be found in Figure 5.6

In addition, the following attribute is being generated in this process:

- *est_cap_M_m3_per_d*: This is the estimated maximum flow of gas volume in the direction of the pipe flow.

In addition, the component *BorderPoints* also supplies information to this process, such as flow direction from one country to another country, and the maximum amount that can be shipped from one country to another country. However, *BorderPoints* are different to *Storages*, where *Storages* are acting as sinks during the summer period and as sources during the winter period. *BorderPoints* however, could be sinks and sources during summer or winter, their real flow behaviour throughout the year is not known. Hence it is assumed, that they can be both, sinks and sources at the same time. The process described next has been implemented to reflect this, and will be explained for the country Spain.

Spain shares 6 elements of type *BorderPoints* with its neighbours. In a first iteration, it was assumed, that all of those elements were directing the gas flow from outside of Spain towards Spain. For this configuration, the gas flow is simulated for the Spanish pipes with the other components, such as *LNGs* and *Storages*. In a second step, one of the border point elements is "switched" so that gas is leaving Spain at that element, and the gas flow is simulated again for Spain. This process is carried out for all combination of flow directions of the border points. In a subsequent step, an average is formed over all those results, which supplies the final values for gas flow directionality for each Spanish pipe.

This process is carreid out for each country, hence resulting in direction information for each pipe line. However, for countries like Germany, which contains 30 elements of type *BorderPoints*, the above described method would lead to 2^30 permutation, too large carry out. Hence, a maximum number of permutations was set to 10,000, where the flow direction at the border points was selected on a random basis.

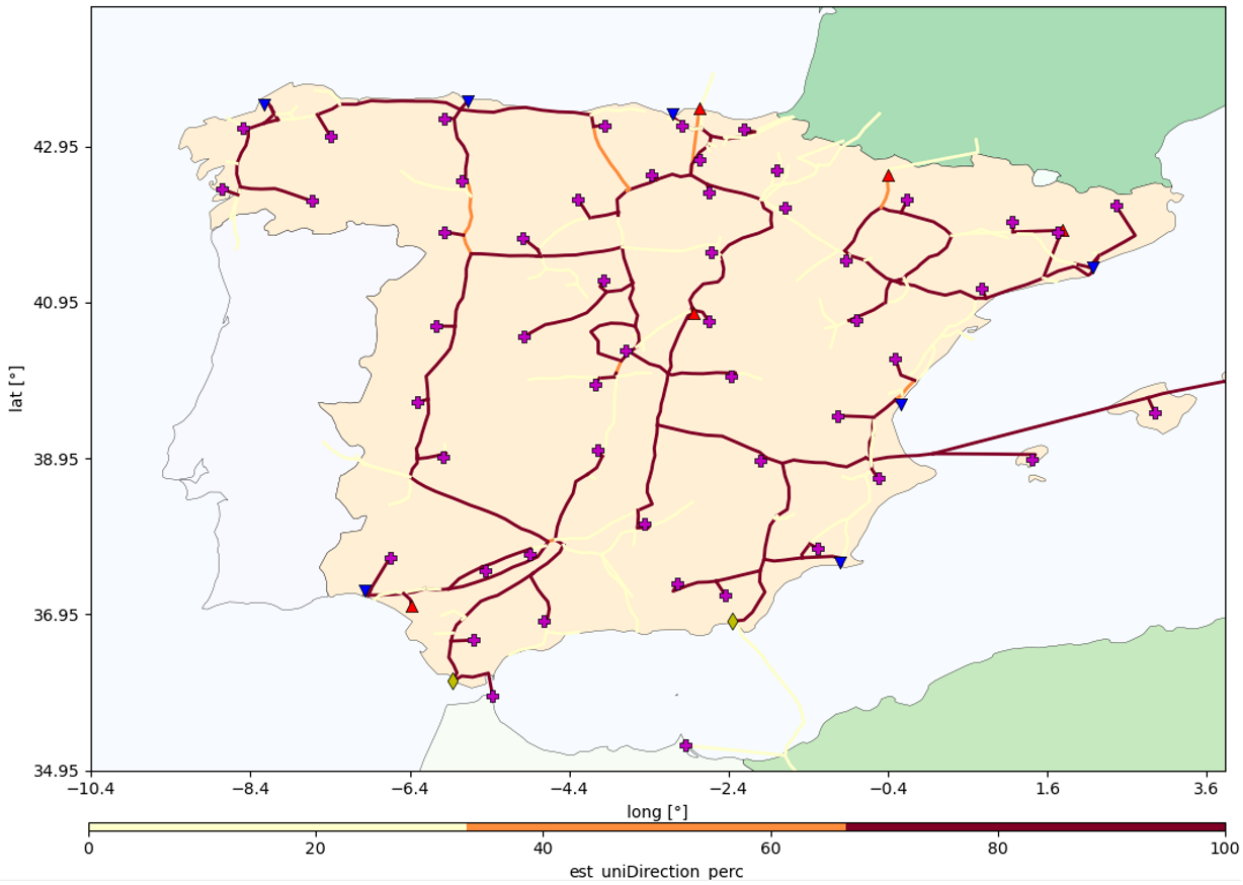


Figure 5.6: Estimated directionality of the pipes in Spain using the “capacity” method.

Requirements for the generation of pipe direction information

As has been described in the above capacity section, the gas demand and supply from the sources and sinks needs to be known, so that the directional information can be determined. The heuristic might not estimate such attribute values for all sinks and sources. Hence, the following gas amounts were assumed as a first cut:

- *Storages*: *max_cap_pipe2store_M_m3_per_d* = 16
- *Storages*: *max_cap_store2pipe_M_m3_per_d* = 30
- *Consumers*: *max_demand_M_m3_per_d* = 4
- *PowerPlants*: *max_demand_M_m3_per_d* = 4
- *LNGs*: *max_cap_store2pipe_M_m3_per_d* = 30
- *Productions*: *max_supply_M_m3_per_d* = 6
- *PipeSegments*: *max_cap_M_m3_per_d* = 44.

However, the user should assume that all those parameters were derived through the general implemented heuristic processes.

5.1.2 Connection point to consumers

During the process of incorporating the GB data set [GB_raw], it was found that only this data set contained any location information for the component *ConnectionPoints*. For the area, for which the GB data set supplied information on pipelines and other components, the GB data set also supplied 147 *ConnectionPoints*. As there were no Pipelines in Northern Island and norther Scottland, there were no *ConnectionPoints* in those regions either.

During the incorporation of the CONS data set at Nuts-3 level, it was found that for the same area that is covered by the GB data set, the CONS supplied 179 Nuts-3 regions. As those to numbers from the CONS and the GB data set are fairly similar, it is assumed that those *ConnectionPoints* from the GB are pipeline outlets, to distribution networks. The SciGRID_gas project does not incorporate any distribution network information, but the closest this data set has are the *Consumers*. Hence it is assumed that the GB *ConnectionPoints* are "equal" to the Cons *Consumers* elements. Hence instead of trying to estimate *ConnectionPoints* for all the other countries in Europe, the roughly 1500 Nuts-3 *Consuemrs* elemnts will be used instead. So as soon as the CONS data set is part of the SciGRID_gas transmission network data set, all *ConnectionPoints* will be removed, and the *Consumers* be attached to the network.

5.1.3 Pipe capacity, pipe diameter and pipe gas pressure

In [KKS+17] (Chapter 4.2.2.) a method was introduced that used the diameter of a pipeline, the pressure of a pipeline and the LKD-type class as the input to generate the missing max flow capacity of a pipeline. This was applied to the LKD data set. It would be good, if this process could be applied to the pipelines outside of Germany. However this was not possible, as the LKD-type was only given for the German LKD data set. In addition, it is not possible to link the LKD-type information to the EMAP-type information, as the EMAP data for Germany was a constant value of 2. Hence it was not possible to relate the LKD-type through a sarrogate to the EMAP data set. Hence a more physical based approach will be applied. It is assumed that the max pipe capacity is proportional to max pressure of the pipe and the cross section of the pipe (pipe radius squared). Hence one can use this relationship to estimate missing values by trainng a polynomial model with corresponding input values. Here the following three attributes can be estimated:

$$capacity \simeq diameter^2 * pressure$$

$$pressure \simeq \frac{capacity}{diameter^2}$$

$$diameter \simeq \sqrt{\frac{capacity}{pressure}}$$

Estimation of pipeline capacity

Here pipeline capacity is estimated by using the pipeline pressure and pipeline dimaeter as the independent variables, where a polynomial model of order 1 is formed by forming a pressure * diameter^2 value. For known values of all three attributes, a polynomial model can be trained. For any cases, where the pressure and the diameter of a pipeline were known, and the capacity is not known, the missing capacity value can be estimated. Here the "polyfit" from the NumPy module was used, also returing an uncertainty value of the fitting parameters, which was used to estimate the uncertainty of the estimated capacity. Currently the z0 and z1 values are 2.12 and 5.55e-7 respectively. The associated uncertainty of the paraters are 4.87 and 3.6e-8 respectively. Here the INET data set was used as the training data set, with 74 pairs of values. The polynomial equation with parameters is given as:

$$max\_cap\_M\_m3\_per\_d = x_0 + x_1 * diameter^2 * max\_pressure\_bar$$

Polynomial fits larger than order one were also investigated, while comparing the adjusted r-square values. It was found that the polynomial fit of order 1 gave the best results in light of simplest model approach.

Estimation of pipeline diameter

The estimation of the pipe diameter followed the same pathway as the estimation of the pipeline gas pressure. Currently the polynomial parameters x_0 and x_1 are 4.44e+5 and 1.15+6 respectively. The associated uncertainty of the paraters are 8.90e+4 and 1.03e+5 respectively. Here the INET data set was used as the training data set, with 74 pairs of values, and the following polynomial equation is:

$$diameter = \sqrt{x_0 + x_1 * \frac{max\_cap\_M\_m3\_per\_d}{max\_pressure_bar}}$$

Estimation of pipeline gas pressure

The estimation of the max pressure in the pipeline follows the same pathway as the estimation of the pipe capacity. Currently the x_0 and x_1 values are 51.6 (+/- 6.2) and 6.8e+5 (+/- 1.2e+5) respectively. The corresponding polynomial equation is:

$$max\_pressure\_bar = x_0 + x_1 * \frac{max\_cap\_M\_m3\_per\_d}{diameter^2}$$

5.2 Statistical heuristic processes

The SciGRID_gas project has been set up to generate a data set of the European gas transmission network. Despite merging several data sources of gas facilities, the resulting gas component data set will contain a large number of missing values. This section here describes how missing attribute values can be generated through different heuristic methods.

In this section, the problem of missing data is described with the help of some mock data set. This is followed by the description of the heuristic methods that have been implemented, and the general pathway that the user needs to undertake to eliminate missing values.

Problem description

In Figure 5.7 the data set contains different elements of different components, where many attribute values could not be found. The example given in the following sections shall depict the gas pipelines, where the attributes in question are *diameter*, *capacity* and *pressure*. The data is summarized in Table 5.1.

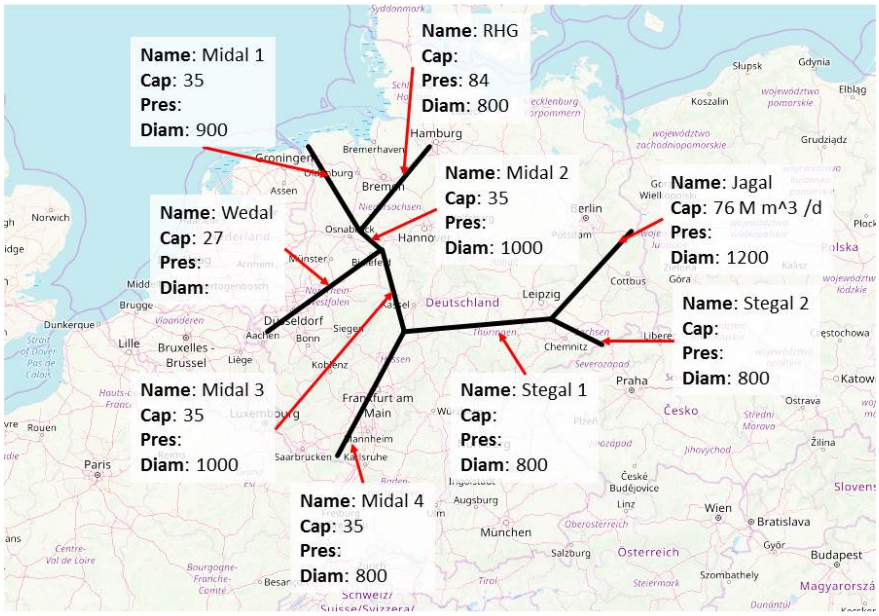


Figure 5.7: Map of some of the larger pipelines in Germany, with corresponding attributes *capacity* (Cap), *pressure* (Pres), and *diameter* (Diam).

Table 5.1: Summary of data of the nine sample pipelines from Figure 5.7.

Pipeline name	*capacity* [M m ^{3} d $^{-1}$]	*pressure* [bar]	*diameter* [mm]
Jagal	76	80	1200
RHG		84	800
Midal 1	40		900
Midal 2	50		1000
Midal 3	35		800
Midal 4	34		800
Stegal_1			800
Stegal_2			900
Wedal	27		

As can be seen, all but one *PipeSegments* contain an attribute value for the attribute *diameter*. For the attribute *capacity* three values are missing, and of all nine *PipeSegments*, only two have a value for the attribute *pressure*. The corresponding data densities for the attributes *capacity*, *pressure* and *diameter* are 67 %, 22 % and 89 % respectively. The overall goal will be to achieve a data density of 100 % for all attributes.

5.2.1 Fill value methods

As one can see, the *capacity* attribute value is given for six of the nine facilities. Several options exist in determining the missing values. A simple solution would be to use the average or median of the input values as a method of estimating any missing value. Here the **mean** and **median** value are $41\,\mathrm{Mm^3 d^{-1}}$ and $35\,\mathrm{Mm^3 d^{-1}}$, respectively. However, selecting the best approach can be difficult, and needs to be transparent. Hence, an "estimation uncertainty" term will be used as decision criteria to determine the best method.

Conventionally, in the worlds of data engineers and big data, one splits the data into a training data set and a test data set. Normally a 70/30 rule is applied, where 70 % of the data ends up in the training data set, and 30 % in the test data set. In the first step a method (e.g. **median**) is applied to the training data set. In the second step, the fitted method results are used to predict the values for the test data set. In the third step one calculates the absolute difference between the method results and the original test data set values (absolute error). The smaller the absolute error, the better the method. This error value could be used as the "estimation uncertainty" and can be used to choose the method that would estimate any missing values.

However, the SciGRID_gas project only contains relatively small data sets. Any splitting of the input data set into training and test data sets would create a data set too small for training and testing purposes. As an example, there are roughly 35 *LNGs* terminals in Europe, and splitting such data set would result in roughly 10 values for testing purposes only. Hence, throughout the SciGRID_gas project the "Leave-one-out" method will be used (see Chapter 10.9), and the error is calculated using the "mean absolute error" (**MAE**), where the absolute error is the absolute difference between a single raw input data value and the model estimation of that value instance. This means, instead of having a 70/30 percent split, one uses all but one data value for training the model and then uses the trained model to estimate the one data value that was not part of the training process. This is being repeated for all data values.

The **MAE** for the **mean** and the **median** method is $25\,\mathrm{Mm^3 d^{-1}}$ and $16\,\mathrm{Mm^3 d^{-1}}$, respectively. Hence, based on the **MAE**, it would be best to use the **median** method approach, and one could fill all missing values with the value of $35\,\mathrm{Mm^3 d^{-1}}$, with an **MAE** of $16\,\mathrm{Mm^3 d^{-1}}$. The **median** method is normally selected for data sets, which contain outliers or the data is not normal distributed. However, the sample size is small, and one could argue, to select the method with the smallest **MAE**. However, overall, the **MAE** is very large in respect of the actual *capacity* value. Therefore, other method approaches also need to be investigated.

An attribute could also be related to other attributes. Here, one could use a linear regression. However, linear regressions tend to weight the independent feature data equally, if more than one is given. Other methods, such as the Lasso-linear regression, tend to weight the independent variables unequally and can even indicate that it would be better to remove some independent variables [Wik20d]. Therefore, if not stated otherwise, the Lasso-linear regression will be used here, instead of a simple linear regression.

Here, the Lasso-linear regression is applied to the *capacity* variable (also referred to as the "predictor" or "regression input"), and the variable *diameter* is the independent variable (also referred to as the "feature" variable). For the pipe RHG, where the *capacity* value is missing and a *diameter* value of 800 mm is given, the Lasso-linear regression estimated the following *capacity* value: $33.9\,\mathrm{Mm^3 d^{-1}}$ with a **MAE** of $2.4\,\mathrm{Mm^3 d^{-1}}$. As one can see, the **MAE** of the Lasso method is significantly smaller when compared with the **MAE** of the **mean** and the **median** methods. However, this example should not lead to the assumption that a Lasso-linear regression is always better than a simple estimation using a **mean** or a **median** value. For example, an attribute data set could be unrelated to any other attribute; hence, using a Lass method would be wrong. In addition, for some attribute values the methods of **mean** or **median** might have to be used, due to lack of feature data. Here, the Lasso-linear regression was implemented in the same way that the **mean** and **median** methods were implemented, using the "Leave-one-out" method.

The process described above has been applied to the example data presented in Table 5.1. Here Table 5.2 and Table 5.3 summarize the input values, estimation values, estimation method, and the corresponding **MAE** based on the "Leave-one-out" approach. Results for the attribute *capacity* are given in Table 5.2, and results for the attribute *diameter* are represented in Table 5.3. As the attribute *pressure* only contained two input values, no values could be estimated with the above "Lasso" process, as the system has been set up that it needs at least four values. For the other two attributes, the input and estimated values are being presented, and the difference between estimated and input value is close to the given uncertainty. As one can see, the estimated values agree better with the input data for the method of "Lasso", when compared with the method of "mean". However, not all values could be estimated using the Lasso method, due

to missing values (e.g. *diameter* for pipeline Stegal_2). Hence for some missing values the system used **mean** or **median** method instead (e.g. *capacity* for the pipe "Wedal").

Table 5.2: Input and estimated *capacity* data of the example, including the method of estimation and the corresponding estimated error. Values are given in units of [M m ^{3} d $^{-1}$].

Pipeline name	Input *capacity*	Estimated *capacity*	Method	Uncertainty
Jagal	76	69.7	Lasso	2.4
RHG		33.9	Lasso	2.4
Midal 1	40	42	Lasso	2.4
Midal 2	50	51.8	Lasso	2.4
Midal 3	35	33.9	Lasso	2.4
Midal 4	34	33.9	Lasso	2.4
Stegal_1		33.9	Lasso	2.4
Stegal_2		42.8	Lasso	2.4
Wedal	27	43.7	Mean	12.9

Table 5.3: Input and estimated *diameter* data of the example, including the method of estimation and the corresponding estimated error. Values are given in units of [mm].

Pipeline name	Input *diameter*	Estimated *Diameter*	Method	Uncertainty
Jagal	1200	1233	Lasso	23
RHG	800	900	Mean	100
Midal 1	900	875	Lasso	23
Midal 2	1000	975	Lasso	23
Midal 3	800	826	Lasso	23
Midal 4	800	816	Lasso	23
Stegal_1	800	900	Mean	100
Stegal_2	900	900	Mean	100
Wedal		746	Lasso	23

Hopefully the above example and description can be used as an explanatory blueprint of the problem that the SciGRID_gas project is facing, and how the missing value generation has been approached. The following section will describe the implemented method pathway within the SciGRID_gas project code.

5.2.2 Attribute value generation pathway

This section describes how the generation of the missing attribute values has been implemented into the SciGRID_gas project. Overall, there are six steps that need to be carried out in chronological order. They are described in more detail in the following sub sub-sections, and listed here:

1) Loading network data
2) Configuration of the setup files
3) Generation of plots for data quality and process assurance
4) Parameters generation for the heuristic methods
5) Selecting individual estimation methods for each attribute
6) Simulation of missing attribute values.

1) Loading network data

The first step is to load the data into memory. Functions have been designed as part of the SciGRID_gas project, and will be introduced in an upcoming documentation.

2) Configuration of the setup files

In the next step the user needs to set up the three required setup files. The first one ("Copy_Attribs.csv") allows to copy attribute values from one element to another element, where the elements are of different component type. The second setup file ("StatsMethodsSettings.csv") contains meta information for each method (e.g. **mean**, **median**). The third setup file ("StatsAttribSettings.csv") contains a list of attributes, including attribute-specific metadata. All three setup files are being described next and are located in the following folder "../Ausgabe/GeneratedNetz/Default_SetupFiles/".

Copying_Attribs.csv

Here the user has the option of copying values from one type of component to another type of component, as long as the elements are physically connected to each other. This option has been implemented, as the attribute *max_power_MW* from the component *Compressors* might be "related" to the attributes *max_pressure_bar*, *diameter_mm* and *max_cap_M_m3_per_d* of the component *PipeSegments*. However, current methods implemented do not allow for a correlation across different components. Hence, a method has been implemented that copies attributes from one component to another component, as long as the elements are physically connected. In the above example the *Compressors* attribute *max_power_MW* could be moved to those *PipeSegments* elements that are physically connected to the *Compressors* element (having the same *node_id*). An example of the "Copying_Attribs.csv" setup file is given in Figure 5.8.

	A	B	C	D	E
1	Comp_Source	Attrib_Source	Comp_Destination	Attrib_Destination	FillMethod
2	PipeSegments	pipe_class_EMap	Compressors	Pipe_pipe_class_EMap	fill
3	PipeSegments	diameter_mm	Compressors	Pipe_diameter_mm	fill_max
4	PipeSegments	max_cap_M_m3_per_d	Compressors	Pipe_max_cap_M_m3_per_d	fill_max
5	PipeSegments	min_cap_M_m3_per_d	Compressors	Pipe_min_cap_M_m3_per_d	fill_min
6	PipeSegments	max_pressure_bar	Compressors	Pipe_max_pressure_bar	fill_max
7	PipeSegments	pipe_class_type	Compressors	Pipe_pipe_class_type	fill
8	PipeSegments	max_cap_M_m3_per_d	LNGs	Pipe_max_cap_M_m3_per_d	fill_max
9	PipeSegments	pipe_class_type	LNGs	Pipe_pipe_class_type	fill
10	PipeSegments	pipe_class_EMap	LNGs	Pipe_pipe_class_EMap	fill

Figure 5.8: Sample of the file "Copying_Attribs.csv".

The file contains the following five columns:

- **Comp_Source**: String, name of the component type, from where the data shall be from.
- **Attrib_Source**: String, name of the attribute type, containing the data to be used.
- **Comp_Destination**: String, name of the component type, to where the data shall be written to.
- **Attrib_Destination**: String, name of the attribute type, into which the data shall be written in to.
- **FillMethod: String, containing one of the following options:**
  - **fill**: Copies variable value to destination element, if no value was given in the destination element.
  - **fill_max**: Copies variable value to destination. In addition, checks if destination value smaller than current source value and if so then replaces with new source value. Otherwise will fill destination attribute.

  - **fill_min**: Copies variable value to destination. In addition, checks if destination value is larger than current source value and if so then replaces with new source value.
  - **replace**: Copies variable value to destination element, and overwrites any existing value.

These new attributes can be used throughout the heuristic attribute generation process. Subsequent to the heuristic processes, those generated values and attributes will be removed prior to the release of the data set.

StatsMethodsSettings.csv

In this file the user selects which methods (e.g. "Lasso") shall be used for testing the data and their relationships. A sample method setup file is given in Figure 5.9.

	A	B	C
1	MethodNam	Param	ToBeApplied
2	Lasso		1
3	LogisticReg	{'solver':'lbfgs'}	1
4	Mean		1
5	Median		1
6	Min		1
7	Max		1

Figure 5.9: Sample file of the file "StatsMethodsSettings.csv".

Column "A", which has the label "MethodName", contains the heuristic method names that have been implemented into the SciGRID_gas project code. Currently the following methods have been implemented:

- **Lasso**: A type of linear regression that uses shrinkages. It has been described as [sl19]: "The Lasso is a linear model that estimates sparse coefficients. It is useful in some contexts due to its tendency to prefer solutions with fewer non-zero coefficients, effectively reducing the number of features upon which the given solution is dependent. For this reason, Lasso and its variants are fundamental to the field of compressed sensing. Under certain conditions, it can recover the exact set of non-zero coefficients."
- **LogisticReg**: Here the attributes to be predicted are of binary type or are multiple discrete values. The Logistic-regression is described by the scikit-learn.org web portal as [sl19]: "Logistic regression, despite its name, is a linear model for classification rather than regression. In this model, the probabilities describing the possible outcomes of a single trial are modelled using a logistic function.".
- **Mean**: Calculation of the **mean** attribute value of the predictor attribute values.
- **Median**: Calculation of the **median** value of the predictor attribute values.
- **Min**: Calculation of the **min** value of the predictor attribute values.
- **Max**: Calculation of the **max** value of the predictor attribute values.
- **OLS**: Calculation using the linear "Ordinary Least Squares" method[1].
- **Logarithmic**: Calculation using logerithmic input values. Otherwise, it is the same process as described under **Lasso**.
- **OneOver**: Calculation using the oen over input values. Otherwise, it is the same process as described under **Lasso**.
- **Squared**: Calculation using the squared input values. Otherwise, it is the same process as described under **Lasso**.

[1] https://www.statsmodels.org/stable/generated/statsmodels.regression.linear_model.OLS.html

The second column with the label "Param" contains possible parameters that are applied to the method within the SciGRID_gas Python code. Here for the **LogisticReg** a solver is needed to be specified. As can be seen, the following entry was supplied:"{'solver':'lbfgs'}" (See [Wik20e] for an explanation of the 'lbfgs' solver). All other methods currently do not need additional parameter settings.

The column "ToBeApplied" describes if the method should be a part of the test and determination suit ("1") or not ("0").

StatsAttribSettings.csv

In this CSV file (**StatsAttribSettings.csv**), additional information in respect of the attributes is being supplied. Here, the user selects the attributes (e.g. *max_cap_M_m3_per_d*) that shall be used during the heuristic testing suit. A sample of such a file is presented in Figure 5.10.

	A	B	C	D	E	F
1	CompName	AttribName	Features	Predictors	Convert2Float	RegressionType
2	Compressors	max_cap_M_m3_per_d	1	1	0	lin
3	Compressors	is_H_gas	1	1	0	log
4	Compressors	max_power_MW	1	1	0	lin
5	Compressors	max_pressure_bar	1	1	0	lin
6	Compressors	num_turb	1	1	0	lin
7	Compressors	turbine_fuel_isGas_1	1	1	0	log
8	Compressors	turbine_fuel_isGas_2	1	1	0	log
9	Compressors	turbine_fuel_isGas_3	1	1	0	log

Figure 5.10: Sample of the file "StatsAttribSettings.csv".

The file consists of six columns and they are described as follow:

- **CompName**: This column contains the component name. Options are all component names as introduced in Chapter 2.
- **AttribName**: This column contains the attribute names of the component given under "CompName" that can be part of the heuristic testing process.
- **Features**: This indicates that the attribute values shall be a feature variable ("1") or not ("0").
- **Predictor**: This indicates if this variable shall be tested ("1") or not ("0"). Attributes with value settings of "1" will be loaded, independent of the settings under "Feature".
- **Convert2DiscreteValue**: This indicates, if the loaded data is to be converted from string variables to numbers. Below in Figure 5.11 an example is given of a column of "gender" entries and "age" values, where the attribute *gender* is being converted.

RegressionType This is a string, indicating the regression method to be applied to the data. The following options are currently implemented:

- "lin": This stands for "linear regression", and includes the Lasso linear regression, median, min and max sample values.
- "log": This stands for "logistic regression", and refers to a logistic regression.

Hence, with the above settings, the user can supply all information required for the testing phase, and the user has the option of modifying the testing runs by excluding certain variables. The following section will describe further required steps the user will need to undertake as part of the attribute value generation.

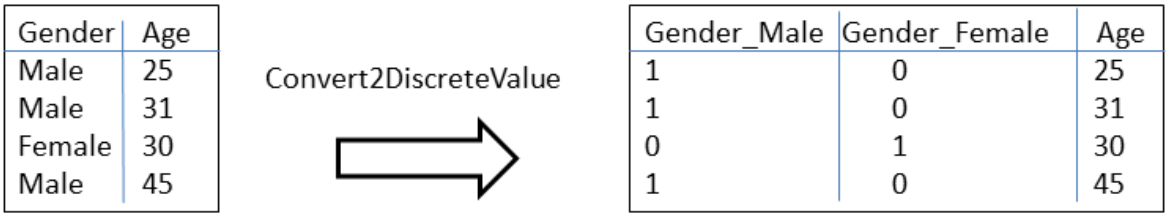


Figure 5.11: Example of converting string attributes to number attributes.

3) Generation of plots for data QA

Having a good understanding of the quality and quantity of the data is very important, before one can apply any heuristic methods for generating missing attribute values. Hence, **THE USER NEEDS TO (VISUALLY) INSPECT THE DATA THAT CAN BE USED AS INPUT FOR THE HEURISTIC PROCESSES!**

This is carried out with the function **M_Stats.gen_DataHists(** *Netz, CompNames, AttribNames, StatsInputDirName, DataStatsOutput* **)**. It requires the following inputs:

- *Netz*: A copy of the network.
- *CompNames*: A list of components to be visualized.
- *AttribNames*: A list of attribute names to be visualized.
- *StatsInputDirName*: A relative path to the above setup file "StatsAttribsSettings.csv".
- *DataStatsOutput*: A relative path to the main folder, where the plots will be stored into. After the plot generation, this folder will contain the following:
  - **HistPlots**: A subfolder containing further subfolders, one for each component and each subfolder will contain the corresponding plots. Each plot consists of a scatter-plot of the data, and a histogram, see Figure 5.12 as an example. In addition, the title contains the name of the attribute next to information in respect of the attribute data density.
  - **Overview.png**: A file with the name "Overview.png" and contains pair-plots of attributes. (See Figure 5.13 as an example). Here each attribute of a component is plotted against all other attributes of the same component. This can be used to investigate if there are correlations between individual attributes.

With the help of the plots and data density values, the user can embark on the following steps:

- Correct any wrong data.
- Adding more data where data density is low, if possible.
- Select and unselect attributes from the subsequent steps due to data distribution and data density issues, resulting in changes in the setup files.

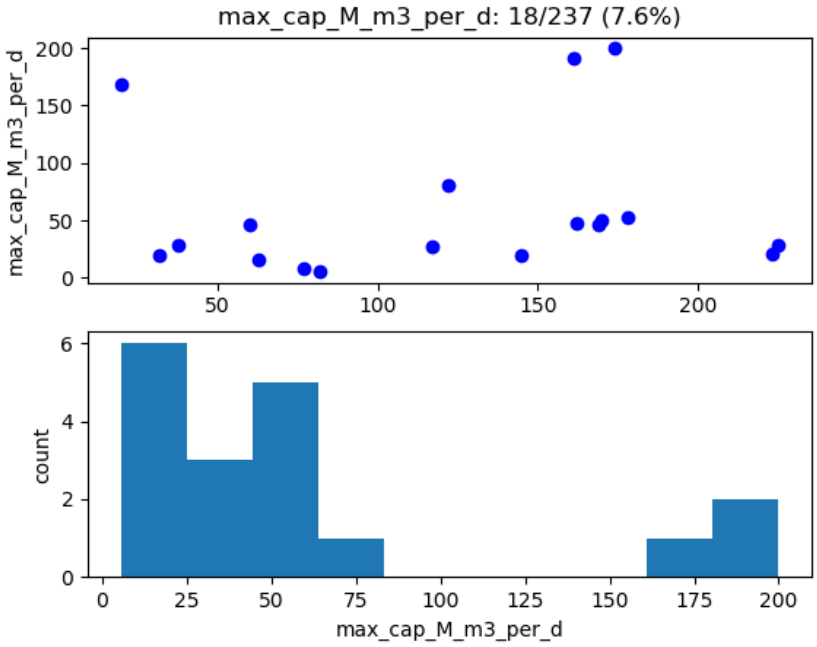


Figure 5.12: Example histogram plot of the *Compressors* attribute *max_cap_M_m3_per_d*.

4) Parameters generation for the heuristic methods

After the data has been loaded into the Python memory and the setup files have been adjusted, the methods in conjunction with the feature attributes will be used to test to generate missing values (predictors). For this the function **M_Stats.gen_StatsParam(** *Netz, CompNames, StatsInputDirName, DataStatsOutput, MaxCombDepth* **)** has been generated. It needs the following inputs:

- *Netz*: A copy of the gas component data set.
- *CompNames*: A list of component names for which this process needs to be carried out.
- *AttribNames*: A list of attribute names for which this process needs to be carried out.
- *StatsInputDirName*: A relative path, where both input setup files can be found.
- *DataStatsOutput*: A relative path, where output information will be written to.
- *MaxCombDepth*: This gives the number of independent attributes, which can be used by each estimation method. The larger this value, the more combinations exist for a given list of independent attributes. However, more significant is that larger "MaxCombDepth" can lead to over-fitting. The number of resulting combinations of attributes can be estimated using $n!/(r!(n-r)!)$, where n the number of attribute variables to choose from, and r of them are chosen, where repetition is not allowed and order does not matter.

The output of the function **M_Stats.gen_StatsParam()** will be twofold: additional plots and measures of fit values.

A sample of such a plot is given in Figure 5.14. Each predictor (here *max_cap_M_m3_per_d*) is plotted against the selected features that were used to determine the predictor attribute (here *max_power_MW*). Here the predictor is plotted on the y-axis, whereas the feature is plotted on the x-axis. The solid line is the estimation of the method used. The title contains information on the method selected, and an R-square value of the fit that was determined using the method.

In addition to the graphical output, additional simulation information is stored in individual CSV files for each component. These files are described in more detail below.

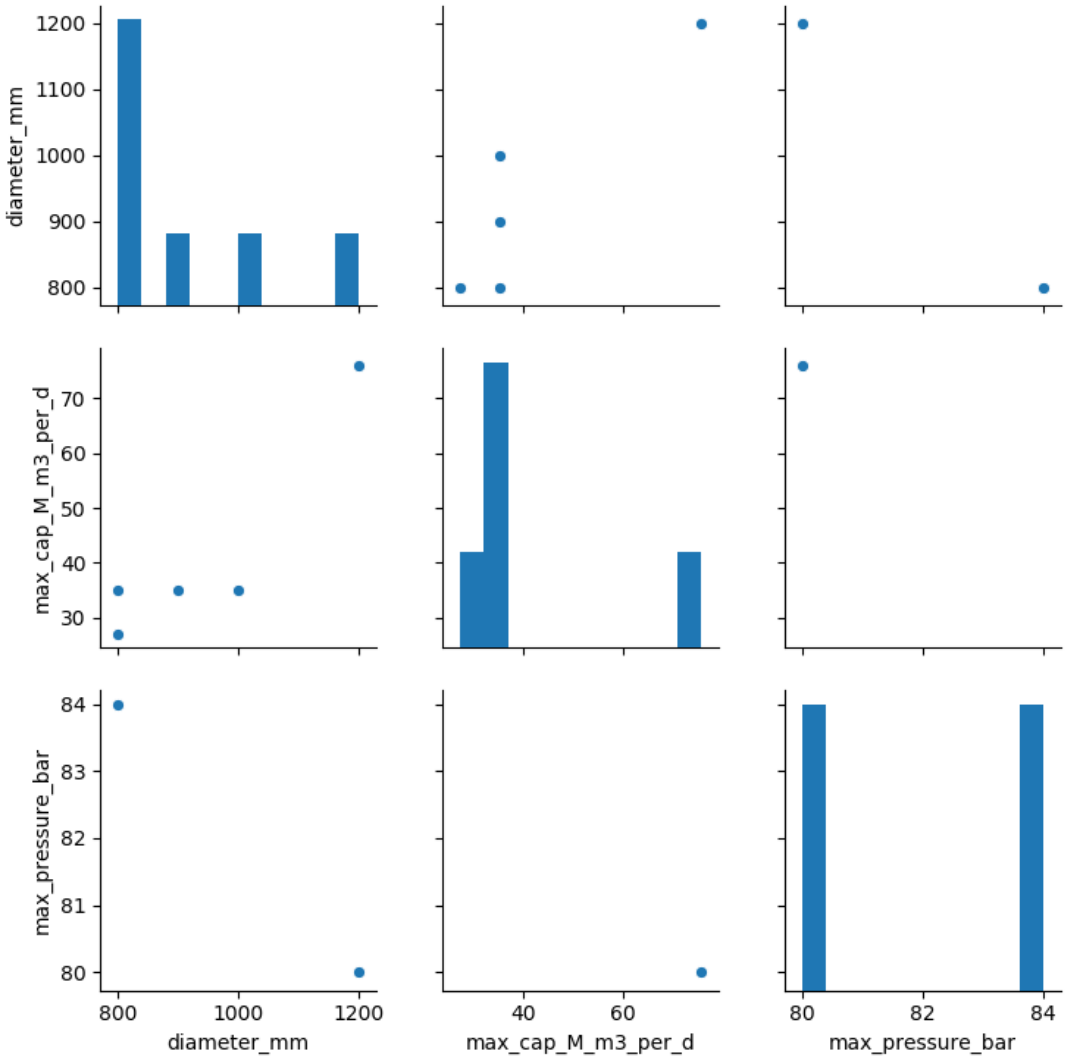


Figure 5.13: Overview of the mutual attribute relations for the component *Compressors*.

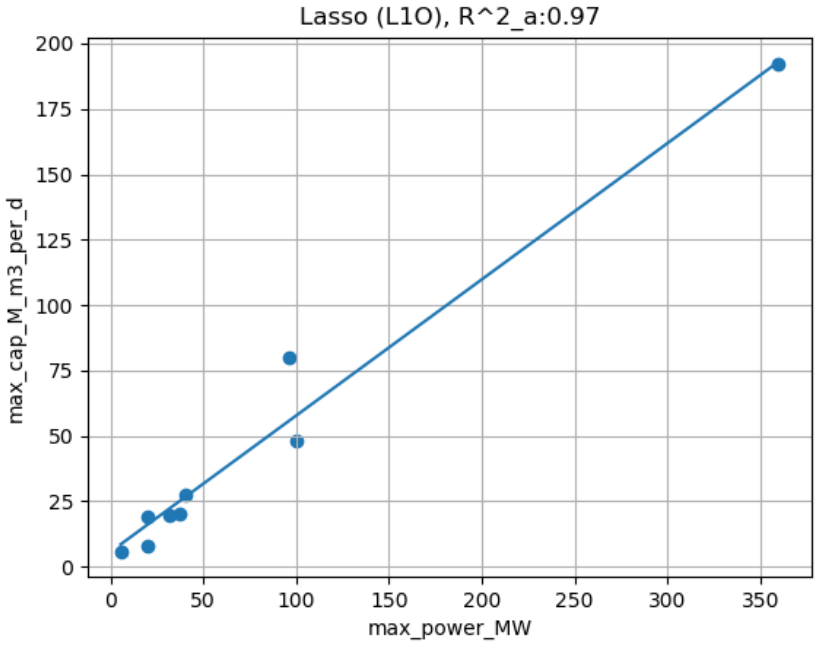


Figure 5.14: Example of attribute *max_power_MW* versus *max_cap_M_m3_per_d* from the component *Compressors*. The solid line represents the fit of the Lasso method to the data.

An example output file is given in Figure 5.15 and Figure 5.16 for the component *LNGs*.

	A	B	C	D	E	F
1	CompName	AttribName	NumElements	ModelName	NumFeatures	FeatureNames
2	LNGs	max_cap_store2pipe_M_m3_per_d	32	Lasso	1	["Pipe_max_cap_M_m3_per_d"]
3	LNGs	max_cap_store2pipe_M_m3_per_d	32	Lasso	1	["median_cap_store2pipe_M_m3_per_d"]
4	LNGs	max_cap_store2pipe_M_m3_per_d	32	Lasso	1	["max_workingGas_M_m3"]
5	LNGs	max_cap_store2pipe_M_m3_per_d	32	Mean	1	["max_cap_store2pipe_M_m3_per_d"]
6	LNGs	max_cap_store2pipe_M_m3_per_d	32	Median	1	["max_cap_store2pipe_M_m3_per_d"]
7	LNGs	max_workingGas_M_m3	32	Lasso	1	["median_cap_store2pipe_M_m3_per_d"]
8	LNGs	max_workingGas_M_m3	32	Lasso	1	["max_cap_store2pipe_M_m3_per_d"]
9	LNGs	max_workingGas_M_m3	32	Mean	1	["max_workingGas_M_m3"]
10	LNGs	max_workingGas_M_m3	32	Median	1	["max_workingGas_M_m3"]

Figure 5.15: Example CSV output of heuristic model results for the component *LNGs*, depicting columns A - F.

These files are written to the folder "../Ausgabe/GeneratedNetz/. . . /StatsData/". All generated files start with the name "RetSummary" and are followed by the name of the component, separated by an underscore. Therefore, the CSV file name for the component *LNGs* is "RetSummary_LNGs.csv".

The files contain information on attributes, methods, errors and parameter settings, where each line is a single run/test result. The columns are as follow:

- *CompName*: Name of the component.
- *AttribName*: Name of the predictor attribute.
- *NumElements*: Number of elements of this component.
- *MethodName*: Name of the method used that was selected through the setup file "StatsMethodsSettings.csv".
- *NumFeatures*: Number of features used to estimate the predictor.

G	H	I	J	K	L	M	N	O
Plots	NumSamples	NumFill	BIC	MeanAbsError	R_2	R_2_adj	ReplaceType	ModelParam
../StatsDa	4	0	14.3057177	4.845552454	0.72296201	0.584443015		{"SC_Mean": [34.5343801375], "SC_Scale": [28
../StatsDa	21	0	85.2509268	4.756034276	0.822467826	0.813124027		{"SC_Mean": [29.326994301428574], "SC_Scale
../StatsDa	28	4	109.381584	5.140814158	0.846444198	0.840538205		{"SC_Mean": [256364758.0], "SC_Scale": [1687
../StatsDa	29	3	166.939796	12.85032496	0	-0.037037037		{"SC_Mean": [0], "SC_Scale": [0], "Intercept":
../StatsDa	29	3	168.158632	12.42324043	-0.04292456	-0.081551394		{"SC_Mean": [0], "SC_Scale": [0], "Intercept":
../StatsDa	21	0	764.014679	69016010.55	0.786511086	0.775274828		{"SC_Mean": [29.326994301428574], "SC_Scale
../StatsDa	28	2	1011.07223	56118773.21	0.849922646	0.84415044		{"SC_Mean": [24.677103718199607], "SC_Scale
../StatsDa	31	1	1180.27533	149945270.6	0	-0.034482759		{"SC_Mean": [0], "SC_Scale": [0], "Intercept":
../StatsDa	31	1	1181.666	147160879	-0.04588166	-0.081946546		{"SC_Mean": [0], "SC_Scale": [0], "Intercept":

Figure 5.16: Example CSV output of heuristic model results for the component *LNGs*, depicting columns G - O.

- *FeatureNames*: List of feature attribute names that were used to estimate the predictor.
- *Plots*: A link to the individual plots of the features and attribute relationship, where the hyperlink currently works under Excel on Windows only.
- *NumSamples*: Number of samples of the feature data, which were used as part of this method evaluation (this number can never be larger than the value in column *NumElements*).
- *NumFill*: Number of elements for which the predictor attribute can be simulated with the method, where the attribute had missing values. (This value also includes the value given under *NumSamples*.)
- *BIC*: Indicator for the goodness of fit of the model using the BIC (Bayesian information criterion) value. The lower the BIC value the better the method fit.
- *MeanAbsError*: Measure of goodness of fit of the model using the mean absolute error (**MAE**).
- *R_2*: R-square model value.
- *R_2_adj*: Adjusted R-square value.
- *ReplaceType*: An empty column that will be used at a later stage.
- *ModelParam*: An entry containing all the fitting parameters used by the methods and attributes, the scaling values of the feature attributes ("SC_Mean", and "SC_Scale"), and the method parameters ("Intercept", "Coef").

With the above information, plots and values, the user can make an informed decision in respect of which method could be used with which attribute to fill missing values.

5) Selecting individual estimation methods for each attribute

As described in the above processes, the function **M_Stats.gen_StatsParam()** generates plots and CSV files containing information on the goodness of the methods for each attribute. With this information the user can decide which method in combination with feature attribute values can be used to generate missing attribute values.

Hence, in the next step the user needs to create a setup file, which contains the settings for methods and attributes that will be used for the generation of those missing attribute values. For this the user can use the output files from the previous step, by carrying out the following actions:

- Copy the above output CSV files to a new location.
- Open one of those files after the other, and carry out the next steps for each file:
  - With the information of the graphs and the indicator of goodness of fit values (e.g. **MAE**), remove all those method attribute combinations that shall not be used for any heuristic processes.
  - Place one of the following keywords into the column *ReplaceType*:

    * "replace": All values will be replaced with the simulated value. Even the original input data will be replaced by the newly simulated values.
    * "fill": Here only the missing attribute values will be determined, therefore, the original input data will not be overwritten, in contrast to option "replace".
    * "fill_ARR": Here missing attribute values are being filled, and values that stem from copyright protected sources are also being overwritten.
    * Order the entries in respect of order of execution, as the methods creating attribute values with the smallest error should be executed first.

However, independent on the replace type setting, some attributes might not be estimated with a single method and single feature attribute set. This can be due to missing feature attribute values required during the estimation process. Hence, the user will need to select several different methods for the generation of all missing attribute values, by retaining several different method lines for a single attribute in the CSV file. This can be seen in Figure 5.15 in the depicted rows two to four. Here, the attribute to be estimated and the model method are the same. However, the feature input variables differ for each line. With this approach one should be able to estimate all missing values. To retain the highest confidence in the estimated values, the user will need to select only those methods and feature attribute value combinations that result in smallest errors.

Here, it is important to notice that the attribute values are generated in the order as they appear in the CSV input file. Hence, the user should order the methods in such a way that the method with the "best" predictions are being carried out first. This could be followed by methods that generate attribute values with larger errors. To assure that there are no further missing values one could retain the **mean** or **median** method as the last method, filling any values that were left unfilled by any previous estimation.

6) Estimation of missing attribute values

The actual estimation and filling of the attributes is carried out with the function **M_Stats.pop_Attribs(** *Netz, CompNames, StatsInputDirName* **)** and is the last step in getting attribute values filled in a gas component data set. This function requires the following input:

- *Netz*: A copy of the component data set.
- *CompNames*: A list of components for which the element's attributes shall be generated and filled.
- *StatsInputDirName*: A relative path name of the location of the above modified setup files.

The return of this function is a component data set, where all missing attribute values have been generated.

5.3 Example value estimation

As an example, a result for the attributes *max_cap_pipe2store_M_m3_per_d* and *max_cap_store2pipe_M_m3_per_d* of the component *Storages* will be presented for the combined IGG data set. The table contains three columns with numbers, with the following definition:

- "N": The number of raw input values; hence, this number is equal or smaller than the number of all facilities of this component.
- "A": The overall average value after all missing values have been estimated, using input and estimated values.
- "MAE": The mean absolute error, of those elements, of which the attribute value had to be determined.

Overall, there are 216 *Storages* elements in the data set. 48 values were missing for both attributes. After the attribute value estimation, the overall **mean()** of the attributes *max_cap_pipe2store_M_m3_per_d* and *max_cap_store2pipe_M_m3_per_d* were 13.9 and 14.6 respectively. The **mean absolute error (MAE)** calculated to be 10.9 and 11.2 for the attributes *max_cap_pipe2store_M_m3_per_d* and *max_cap_store2pipe_M_m3_per_d*. This

seems large in comparison with the overall average value **A**. However, the individual values range from 0 to more than 100 for both attributes.

Table 5.4: List of attributes of the *Storages* component for the IGG data sets, with some statistical properties.

Attribute name	N	A	MAE
max_cap_pipe2store_M_m3_per_d	168	13.9	10.9
max_cap_store2pipe_M_m3_per_d	168	14.6	11.2

A histograms of the raw and the estimated values, depicted in Figure 5.17, gives the distribution for both attributes.

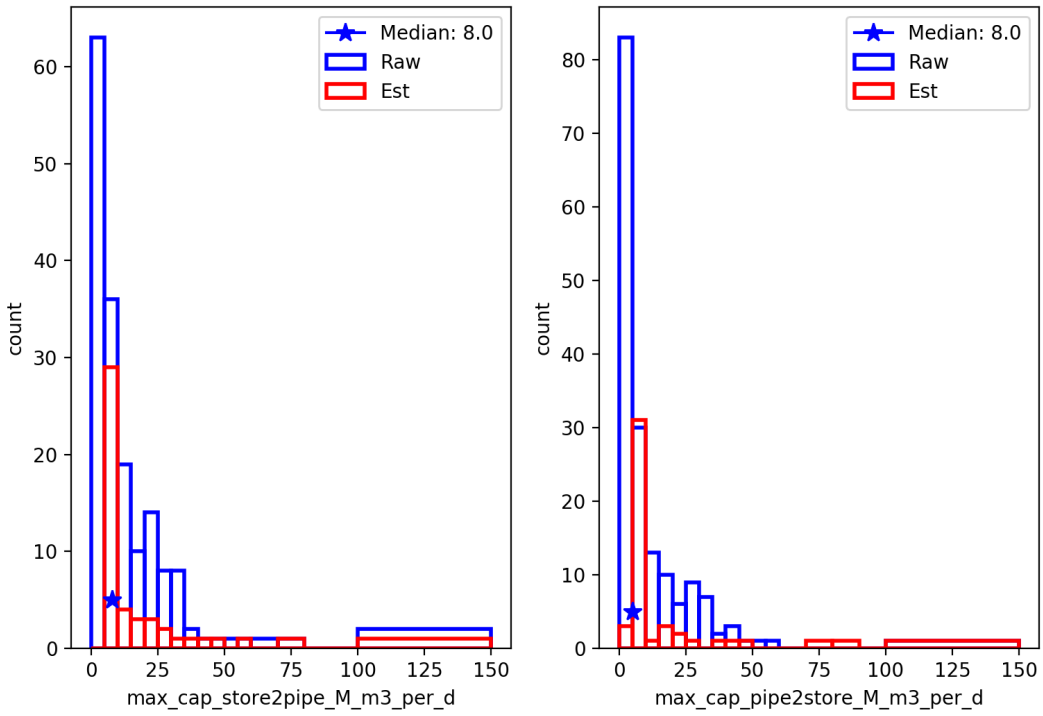


Figure 5.17: Histogram of raw (blue) and estimated (red) values for *max_cap_store2pipe_M_m3_per_d* (left) and *max_cap_pipe2store_M_m3_per_d* (right) of the *Storages* component. Both subplots also indicate the location of the median value for the raw data (star).

The estimated values are roughly distributed the same way that the raw values are distributed. The exceptions are the larger counts of the bin containing the median value of the raw data set. Here the median value is the value used, if there was no other means of determining a missing value. As can be seen, the median value was used substantially for several elements.

Here a quick statistical **Z**-score was carried out [UoO14], giving **Z**-score values of around -1.45 for both attributes. This indicates that the raw and the estimated distributions are the same, as their absolute values are smaller than two. (See Chapter 10.9.6 for a quick introduction to the **Z**-score).

5.4 Automated attribute value generation

In addition to the manual process described in Chapter 5.2 an automated process has also been developed and will be briefly explained below. The automated process has been implemented for users with little statistical background, and to get first results fast.

However, the user should be aware that by applying the automated process, some methods might be selected that lead to incorrect results (e.g. negative *Storages* capacity for increasing max pipeline pressure of connected pipes). Hence, the user should carry out the manual process, instead of relying on the automated process results, to reduce the generation of "bad" attribute values.

To overcome the above-mentioned generation of wrong negative values as part of the automated process, an additional automated process was introduced as described below.

Overall the difference to the manual process described in Chapter 5.2 is that the selection of the attribute generation methods, as described in Chapter 5.2.2, has been automated. As was described in Chapter 5.2.2, one is supposed to select those methods with the best goodness of fit, e.g. **BIC** or **MAE**. Here for the automated process, the **MAE** value is the value that has been selected to determine which attribute generation method is to be used. The automated process selects the methods with increasing **MAE** value. Up to four different methods are automatically selected. In addition, if the median method is not part of the selected methods, then this method will be added to the list of methods to be executed, and will be executed last.

All other processes are as described in Chapter 5.2.

Values supplied with this data set here have been generated using the automated attribute value generation process.

The SciGRID_gas function to execute for the automated process is called **M_AttribHeuristic.getSortedAttribFiles** (*DataStatsInput, DataStatsOutput, AttribValReplaceType = 'fill_arr', AttribMethodSelect = 'MeanAbsError'*), where "DataStatsInput", "DataStatsOutput" are directory path and file name (including a relative path). Here, important is the setting for "AttribValReplaceType", which is set to "fill_arr". This refers to that all missing values will be filled AND all copy right protected values will be overwritten with the estimated value. This assures that the copy right protected data, which was used up to here, will be removed, and the resulting data set can be passed on to external entities. However, if data set to be generated will be used by the user only and not passed on to third parties, then the user can set the value for "AttribValReplaceType" to "fill", where only missing values will be estimated, and the copy right protected data is still part of the data set, which does not break the copy right restrictions.

5.4.1 Attribute bounding box

As described above, some heuristically generated attribute values were negative, or too large to be realistic. Hence, the setup file "../Ausgabe/GeneratedNetz/Default_SetupFiles/AttribBoundaryValues.csv" has been generated, with a screen shot given in Figure 5.18.

As can be seen, there are the following five columns:

- *CompName*: Name of the component, e.g. "Compressors".
- *AttribName*: Name of the attribute, e.g. "max_pressure_bar".
- *MinVal*: Number as lower bound. Any value lower than this given one will be replaced by the value given here.
- *MaxVal*: Number as upper bound. Any value larger than this given one will be replaced by the value given here.
- *UncVal*: Value of the uncertainty that will be written to the data set for any attribute value that was changed through this process here.

Hence, for any corresponding attribute value outside of the *MinVal* and *MaxVal*, the value was changed accordingly and the uncertainty value was changed as well. This assured that all generated values were within the user specified values. Default values have been implemented, and can be found in the above file.

	A	B	C	D	E
1	CompName	AttribName	MinVal	MaxVal	UncVal
2	Compressors	max_pressure_bar	70	200	50
3	Compressors	turbine_power_1_MW	0	50	25
4	Compressors	turbine_power_2_MW	0	50	25
5	Compressors	turbine_power_3_MW	0	50	25
6	Compressors	turbine_power_4_MW	0	50	25
7	Compressors	max_cap_M_m3_per_d	5	200	50
8	Compressors	max_power_MW	2	300	100
9	Consumers	capacity_E_MW	2	5000	1000
10	Consumers	capacity_TH_MW	12	2600	1000
11	Consumers	est_generation_GWh	5	26000	10000

Figure 5.18: Sample of the attribute value bounding box setup file.

5.5 Single network generation

So far raw gas data has been loaded, and converted into SciGRID_gas data sets. The data sets have been combined into a single data set. Any missing attribute values have been generated with the help of implemented automated regression methods.

The last step is to assure that all elements of all components are connected with each other into ONE large network. Here a pathway has been implemented that looks for facilities, such as *Storages* elements that are not connected to a pipeline. Then the closest pipeline is determined, and checked, if the pipeline is closer than a user specified distance. In case that the facility is closer than this distance, the facility is moved to the pipeline. This is carried out with the function **M_Shape.moveComp2Pipe(** *Netz, CompName, PipeName, maxDistance_km* **)**. Any element that is further away than the user specified value is being removed from the final network. The inputs are as following:

Overall, the aggregation was achieved through the following code implementations:

- Removing small sub-networks, if any node of the sub-network is further away from any node of the main network, otherwise those two networks are connected through a pipe. The maximum allowable distance was set to 100 km[1].
- Elements not connected with any pipeline were connected to the network removed (excluding elements of type *Consumers*)[2].
- Some elements of type *Consumers* were not connected to the network. Instead of removing them from the network and losing information on how much gas is being consumed at a NUTS levbel, they were merged with other elements of type *Consumers*, as long as they are located in the same country and one of those elements is connected to the network. Attribute values were added accordingly[3]..
- Pipelines that start and finish at the same node (loop pipelines) were removed, when their length was shorter than 5 km[4].
- Pipelines that were connected at one end only to the network, and the other end was connected to nothing, were

[1] Executed in the code is carried out as demonstrated in part 7.1 of [Die21].
[2] Executed in the code is carried out as demonstrated in part 7.2 of [Die21].
[3] Executed in the code is carried out as demonstrated in part 4.3 of [Die21].
[4] Executed in the code is carried out as demonstrated in part 7.3 of [Die21].

removed[5]. The last step was carried out several times, till no further pipelines could be removed.

- Joining pipelines[6]. Here it is checked if one end of the pipeline is connected only to another end of another pipeline, and then it is checked, that the attribute values of those pipelines allow for a join to take pare. Joints are not allowed, if the two pipelines have too different attribut values, e.g. one pipeline has a *diamter_mm** values of 1000, whereas the other one has a *diameter_mm* values of 1400. The rules outlines in Chapter 4.2 are applied here.

This lead to a significant reduction in pipelines for the European SciGRID_gas gas transmission network data set.

The table below (Table 5.5) shows an example of number of elements prior and after this process, whereas the number of segments and length of overall network did not change in this process. Here the value for *maxDistance_km* was set to 100 km.

Table 5.5: Number of elements prior and post connection with pipelines.

Component name	# of elements prior to process	# of elements post process
BorderPoints	109	109
Compressors	249	248
LNGs	34	32
Storages	302	292
PowerPlants	331	311
Consumers	1506	1367
Productions	109	102
PipeSegments	8145	8386
Nodes	8175	6747

As can be seen, this resulted in discarding several elements, as their distance to the nearest pipeline was larger than the set 100 km. This is quite significant for the component *Consumers*. However a large number of *Consumers* elements were fond in Sweeden and Finland, where gas transmission pipelines were not close by and hence were removed. However, this assures now that the entire data set is a single gas network data set, where all elements are connected with each other and all attribute values have been estimated.

5.6 Summary

Here a method pathway has been described to fill missing attribute values. This is a complex process, where the Python code generates plots and model output values, which needed to be considered by the user. With the information on hand, the user can decide if certain missing attribute values should be estimated using implemented regression methods. In addition, an automated process has been described that can be used to generate all missing attribute values without any additional user input, however, the user has been made aware that this can lead to selecting incorrect attribute relationships, leading to incorrect values, as expert human input is missing.

In addition, a process to generate a single network was briefly introduced, so that all elements, such as LNG terminals, are connected with pipelines with each other. Such generated data set is ready to be used by modellers.

[5] Executed in the code is carried out as demonstrated in part 7.4 of [Die21].
[6] Executed in the code is carried out as demonstrated in part 7.5 of [Die21].

CHAPTER

SIX

DATA AGGREGRATION

Some users might want to use the data in the detailed format as generated for one country or for all of Europe. However for some users this data set might be t detailed and only a maximum number of nodes are required for Europe or a given country. Hence this chapter here will describe how the SciGRID_gas data can be aggregated (reduced in size). These methods can be executed by the end-user with the SciGRID_gas code during or after the final data set has been generated to create a network fit for purpose.

It was found, that there were loop pipelines, starting and ending at the same node. This was observed in the raw LKD data set as well, and they could have a length of up to 6 km due to some of the processes described here. Hence as part of the aggregration process here, any pipeline that started and ended at the same node and had a length of 5 km was removed.

In addition, currently two different aggretation processes have been created. The first merges parallel pipelines into single pipelines. The second method removes pipelines, if they are not connectd to important elements.

6.1 Aggregation of parallel pipes

Most data sets that contain pipelines also contain parallel pipes to some extend, where more than one pipe connects two adjacent nodes. Here several methods will be introduced, where those parallel pipes will be merged into a single pipe, or reduced to a number of parallel pipe smaller than the original number. The method that has been implemented into SciGRID_gas to achieve this is **Agg_PipeSegments()**, which takes the input argument **methodName**. Currently two different methods have been implemented: "SumCap" and "SumNone". Both will be described below.

Here the INET data set for Austria will be used for demonstration purposes. The pipes of component type *PipeSegments* are depicted in Figure 6.1, where the number of paralle pipes between the nodes is also presented. As one can see the number of pipes between pairs of nodes varies from one to three.

6.1.1 Aggregation using "SumCap"

Here a process has been implemented to check for parallel pipes between nodes where conditions have been implemented to assure that the gas flow capacity is treated as the most important attribute. However other pipe attributes like bi-directionality of direction of flow will also be considered. therefore some parallel pipes might be merged whereas others will not be combined. This process has been implemented for pipes of type *PipeSegments* only. However pipes of type *PipeLines* can easily be converted to pipes of type *PipeSegments*.

In the aggregation process of type "SumCap" special emphasis is given to the attribute *max_cap_M_m3_per_d*, and two parallel pipes are only allowed to be merged if one of the following rules are met:

- Both pipes have a value for the attribute *max_cap_M_m3_per_d*.
- Both pipes have no value given for the attribute *max_cap_M_m3_per_d*.

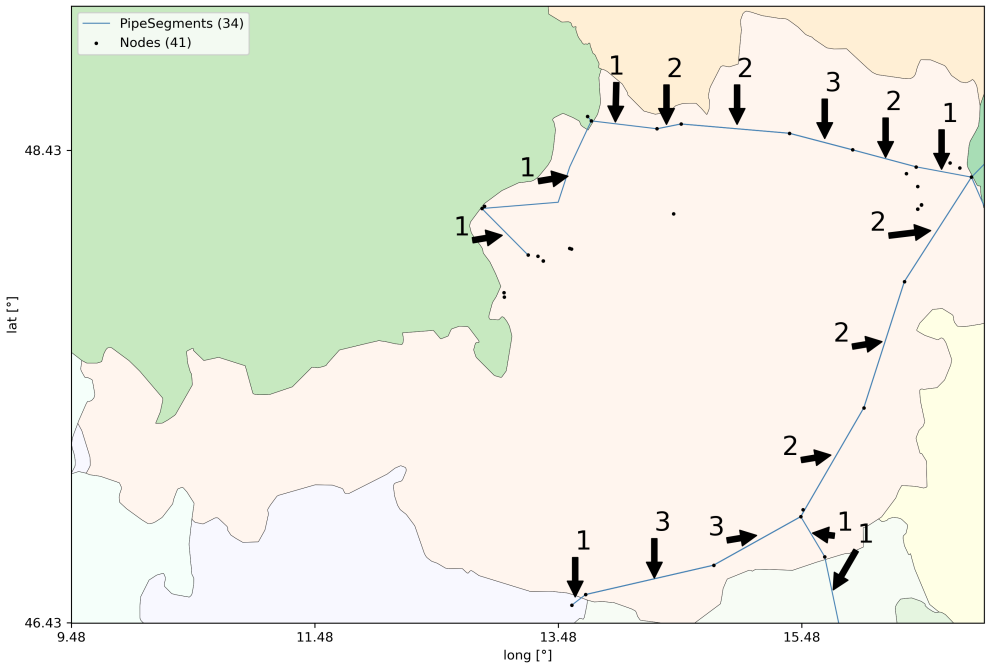


Figure 6.1: Map of *PipeSegments* in Austria, with number of parallel pipes depicted as well.

Hence if one of the pipes has a value given for the attribute *max_cap_M_m3_per_d*, whereas the other one does not, then those two pipes will NOT be merged. This approch for this aggregation method assuers that the information for the maximum capacity of gas flow through pipes is correctly mentained.

In addition, the directionality of the gas flow of each pipe is also being considered under consideration of the attribute *est_uniDirection_perc* presented in Chapter 5.1. The attribute *est_uniDirection_perc* was estimated while investigating the sources and sinks of gas within the gas network, with values from 0 to 100 %. Estimated values between 0 and 33.33 % indicated that the pipe was uni-directional, and gas was flowing from the end node to the start node. For the attribute value larger than 66.66 % it was assumed that the flow is also in one direction only, from the start node to the end node. For all other attribute values (between 33.33 and 66.66 %) the pipe is estimated to be bi-directional. Therefore for the attribute *est_uniDirection_perc*, one of the following options needs to be fulfilled, to allow for the merge process to be continued:

- Both pipes have no information regarding the attribute *est_uniDirection_perc*. Therefore the resulting combined pipe will contain no information for the attribute *est_uniDirection_perc*.
- Only one of the pipelines contains information in respect of *est_uniDirection_perc* (meaning one pipe has a value between 0 and 100, whereas the other pipe has a value of None). The resulting pipe will contain the information of the pipe that contained a numeric value for the attribute *est_uniDirection_perc*.
- Both pipes contain numeric values for the attribute *est_uniDirection_perc*, and both pipe's values fulfil one of the following condition:
  - Both pipes values are smaller than 33.33 % (both pipes have gas flow from from the exit node to the entry node).
  - Both pipes values are larger than 66.66 % (both pipes have gas flow from the entry node to the exit node).
  - Both pipes values are larger than 33.33 % and smaller than 66.66 % (both pipes are bi-directional).

In the case that those two pipes are allowed to be merged, the following steps are being carried out:

- The attribue values for *max_cap_M_m3_per_d* are being added. In the case that no value was given, the None value will be used in the resulting pipe.
- The diameter values of the pipes are added, based on the cross section areas of the original pipes, meaning resulting diameter = sqrt(diameter_pipe_1^2 + diameter_pipe_2^2).
- An average value for the attribute *est_uniDirection_perc* will be estimated from the original merged pipes.
- An average value for the attribute *max_pressure_bar* will be estimated from the original merged pipes.

As stated above, emphasis is put into the attribute of pipe flow capacity, whereas for the other attributes, like diameter or pressure, best possible methods have been implemented, however a few limitations of those attribute combination processes should be listed here. In case that for two parallel pipes only one pipe contains an attribute value for e.g. diameter, whereas the other one does not, then the resulting pipe will contain the information of the pipe, which supplied the value. In a more complicated case, where three pipes are going to be merged into one, and the original pipes conainted the following three attribute values for pressure of 90, None and 120, then the resulting pipe would have the attribute value of 105 (=(90+120)/2).

Results for the example of Autria are given in Figure 6.2 for the number of parallel merged pipes.

In addition, pipeline diameter are presented of the raw input data set in Figure 6.3 and of the aggregated data set in Figure 6.4, where the diameter values are scaled to the same diameter ranges in the plots. In :numref:`Fig_Agg_parall_AT_Raw_diameter' one will only see the information of the last pipe plotted to the graph between nodes, hence overplotting the meta data of the other previously plotted pipe between the same nodes. For the section where there are three parallel pipes (see pipes in northern Niederösterreich), the pipe diameters were 1200, 1200, and 800 mm. After the merge process the single resulting pipe had an estimated pipeline diameter of 1876 mm (based on SQRT(1200^2 + 1200^2 + 800^2)).

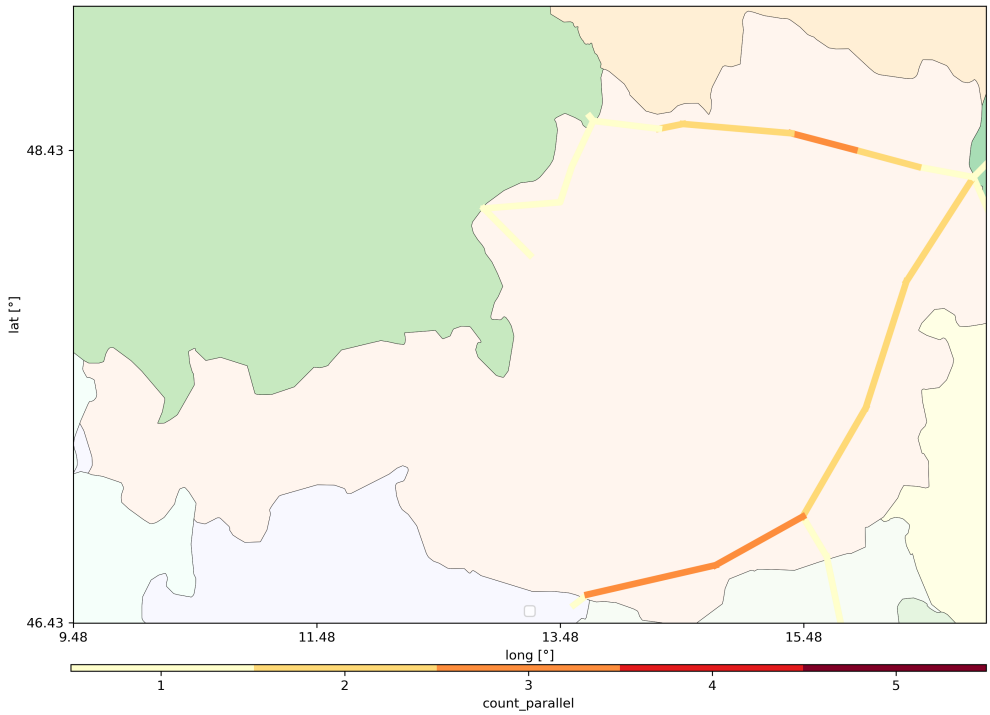


Figure 6.2: Map of the *PipeSegments* for Austria, where the attribute *count_parallel* is depicted, ranging from one to three.

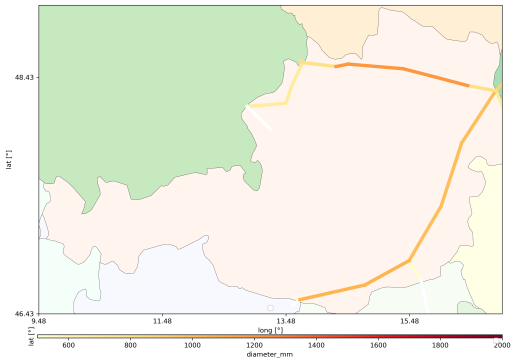


Figure 6.3: Map of the raw *PipeSegments* for Austria, for the attribute *diameter_mm*.

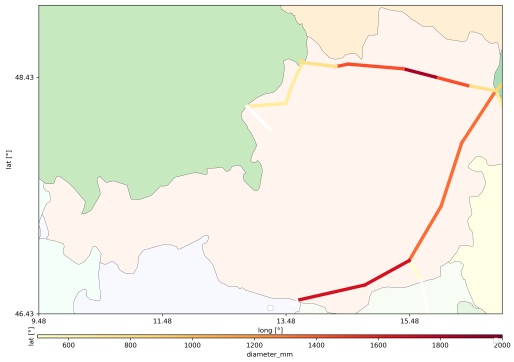


Figure 6.4: Map of the aggregated *PipeSegments* for Austria, for the attribute *diameter_mm*.

6.1.2 Aggregation using "SumNone"

When using the aggregation method "SumNone", the pathway is the same as for the method "SumCap", however no attention is being payed to the attributes *max_cap_M_m3_per_d*, *diameter_mm* and *max_pressure_bar*. So in the case that for two parallel pipes one pipe has a value of None for the attribute *max_cap_M_m3_per_d*, whereas the other has a value of greater than 0, then both pipes will be merged, and the value for *max_cap_M_m3_per_d* of the resulting pipe will be taken from the first pipe. Only the attribute value *count_parallel* will be incremented for each merge.

One can ask why one wants to do such merge, if no attention is being paid to any of the attributes capacity, diameter or pressure? This method was implemented, so that one can visualize the number of parallel pipes with the plotting routine. For the same original data set from above, all parallel lines have been merged, reducing the number of pipes from 34 to 22.

This chapter describes several implemented methods that can be used to aggregate the network. None of the presented aggregations methods have been applied to the final data set.

CHAPTER

SEVEN

FINAL DATA SET

The SciGRID_gas project has the goal of generating a gas transmission network data set for all of Europe. Several individual data sources have been found as part of the project. However, they cannot be used individually, as individual data sets do not contain all the information that is needed for a complete gas transmission network data set. Therefore, several data sets have been combined into a single data set with methods described in previous chapters. After such a process, a significant number of attribute values were still missing in the resulting data set. In [DPS+21] (Chapter Heuristics) described a pathway of how to generate missing attribute values. This resulted in a final gas transmission network data set.

Here the final data set will be described, and differences to a previously published SciGRID_gas data set will be presented as well.

7.1 Combined IGGIELGNC-3 data set

This chapter here will describe the resulting gas transmission network data set, which was constructed by combining the INET, GIE, GSE, IGU, EMAP, LKD, GB, NO and the CON data sets, resulting in the so called IGGIELGNC-3 data set. Each component will be described briefly, mainly focusing on the number of raw versus estimated values. As was described in Chapter 5 two different methods of value estimation have been implemented, a logical/physical based one and a pure statistical based one. It is believed, that summaries should be given for each, so that the reader can get a better understanding of the different methods, and the resulting uncertainties of the approaches.

Here the following terminology in respect of the attribute values will be used:

- "raw": This is referring to the subset of attribute values, that were raw input values.
- "logical/physical": This is referring to the subset of attribute values, that were generated using the logistic/physical methods.
- "statistical": This is referring to the subset of attribute values, that were generated using the pure statistical methods.

7.1.1 PipeSegments

Overall there are 6526 *PipeSegments* elements in the final data set with a length of 236,928 km.

The Table 7.1 depicts the most important attributes that are part of *PipeSegments* elements. The table also presents the number of raw original data, and the number of values that were generated heuristically, including uncertainty values. The table column headings are described below, and will be applicable to all other tables in this chapter here as well:

- "Attribute name": The attribute name.
- "N(R)": The number of raw input values.
- "N(L)": The number of attribute values estimated using a logical/physical base method.

- "N(S)": The number of attribute values estimated using the statistically base method.
- "Ave": Based on the data that is presented in the table, this will be the overall average value of the raw and logical/physical values or the overall average value of the raw and the statistically generated values.
- "Med": Based on the data that is presented in the table, this will be the overall median value of the raw and logical/physical values or the overall median value of the raw and the statistically generated values.
- "U(L)": This is the uncertainty for the logical/physical values.
- "U(S)": This is the uncertainty for the statistical values.
- "Z+(L)": The absolute Z-score (Z+) of the attribute value distributions when comparing the raw distribution with the logical/physical values. An absolute value smaller than two indicates that the distributions are the same.
- "Z+(S)": The absolute Z-score (Z+) of the attribute value distributions when comparing the raw distribution with the statistical values. An absolute value smaller than two indicates that the distributions are the same.
- "P(10)": For a given distribution of values (either raw and logical/physical values or raw and statistical values), the value at the 10 % percentile, informing the user of the spread of the data towards the lower values. (Here no assumptions are being made that the distribution is Gaussian or non-Gaussian, as the determination of the percentile is a simple interpolation of the input values, in respect of the percentile).
- "P(90)": For a given distribution of values (either raw and logical/physical values or raw and statistical values), the value at the 90 % percentile, informing the user of the spread of the data towards the higher values.

Table 7.1: List of attributes of *PipeSegments* elements for the IGGIELGNC-3 data sets, for the raw and logical/physical generated values, with additional statistical properties for each attribute.

Attribute name	N(R)	N(L)	Ave	Med	P(10)	P(90)	U(L)	Z+(L)
diameter_mm	1815	2	859	900	500	1220	139	1.50
is_bothDirection	158	123	0.87	1.00	0.00	1.00	0.10	6.83
max_cap_M_m3_per_d	184	940	35.4	28.8	3.99	69.9	0.32	5.14
max_pressure_bar	1081	104	71.6	70.0	55.0	98.4	1.25	3.88

Table 7.2: List of attributes of *PipeSegments* elements for the IGGIELGNC-3 data sets, for the raw and statistically generated values, with additional statistical properties for each attribute.

Attribute name	N(R)	N(S)	Ave	Med	P(10)	P(90)	U(S)	Z+(S)
diameter_mm	1815	4709	889	900	710	1000	248	5.93
is_H_gas	2783	3743	0.97	1.00	1.00	1.00	0.48	15.2
is_bothDirection	158	0	0.77	1.00	0.00	1.00	N/A	N/A
max_cap_M_m3_per_d	184	5402	28.3	27.4	27.4	27.4	22.8	7.98
max_pressure_bar	1081	5341	70.2	70.0	70.0	70.0	12.7	1.47

Additional attributes that are not supplied, but were part of the attribute generation process are:

- *pipe_class_EMap*
- *pipe_class_LKD*
- *lat_mean*
- *length_km*
- *long_mean*
- *waterDepth_m.*

In addition, other attributes that were part of the data set, but have been removed prior to release are:

- *exact*
- *num_compressor*
- *operator_name*
- *source*.

is_H_gas

The attribute *is_H_gas* has a data density of 43 %, and a high average value **Ave** of 0.97, indicating that a large number of pipelines of the input data set transport high calorific gas. *is_H_gas* is an attribute, for which no relation to any other attribute could be determined. Hence, a constant value of "1" was used to fill all missing attribute values of *is_H_gas*, where an uncertainty of "0.5" was also used for those elements. This approach has been applied to all missing *is_H_gas* attribute values for all components, resulting in very large **Z+**-score value.

max_pressure_bar

The attribute *max_pressure_bar* has a mean value of 70.2, whereas the uncertainty *U* is 12.7 for the statistical approach and an uncertainty of 1.25 of the logical approach. The **Z+**-score for both approaches are above two, indicating that the generated attribute values distribution is slightly different to the distribution of the input data set.

max_cap_M_m3_per_d

The attribute *max_cap_M_m3_per_d* consists of only 184 raw input values. Here the mean absolute error *U* has a value of 22.8 for the statistical approach, whereas the mean value is 28.3, meaning there is a large uncertainty in respect of the attribute values. Here the range of raw input data ranged from a value of 5 to a value of 200. The **Z+**-score is much larger than 2, indicating that the distributions of values between the raw and the generated data sets are quite different, as almost all of the missing values were generated using the median value of the raw input data. However, a large portion of missing values was also generated using the logical heuristic, resulting is much smaller estimated uncertainty and a lower **Z+**-score value.

diameter_mm

This data set contained a larger portion of raw values for the attribute *diameter_mm*, where roughly 28 % of this attribute was supplied as raw values. Here again, a large portion of the missing values were generated by using the median of the raw input values. Better methods need to be implemented. A logistic heuristc was also implemented, however only two missing attribute values could be determined.

Before addressing the other components, information on the distribution of the raw and the estimated values are summarized in Chapter 10.8. This has been carried out through histogram plots. An example of those histogram plots is given in Figure 7.1 for the attribute *max_cap_M_m3_per_d* of the component *PipeSegments*. In addition, the same data is presented on a logarithmic Y-axis scale, so that the smaller bin entries can be seen better (Figure 7.2).

Description of the plots:

- The plot shows in green bars the histogram of the raw input data (left y-axis).
- The red bars indicate the histogram of the estimated values (right y-axis).
- The title contains several items of information:
  - Name of the attribute (excluding the unit)
  - Total number of elements of this attribute

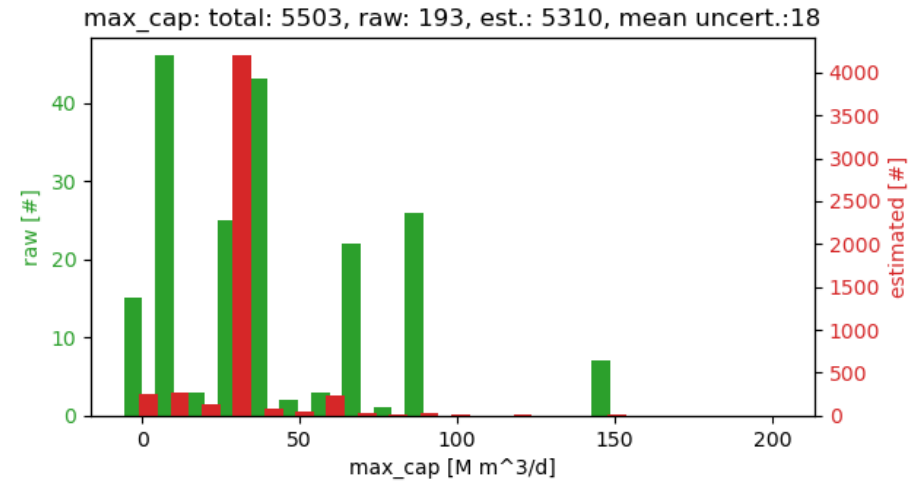


Figure 7.1: Sample plot of the raw and estimated values of the attribute *max_cap_M_m3_per_d* of the component *PipeSegments*. Green bars are the raw input values, red bars are the histogram of the estimated values. The title and the text below the plot are described in the text below.

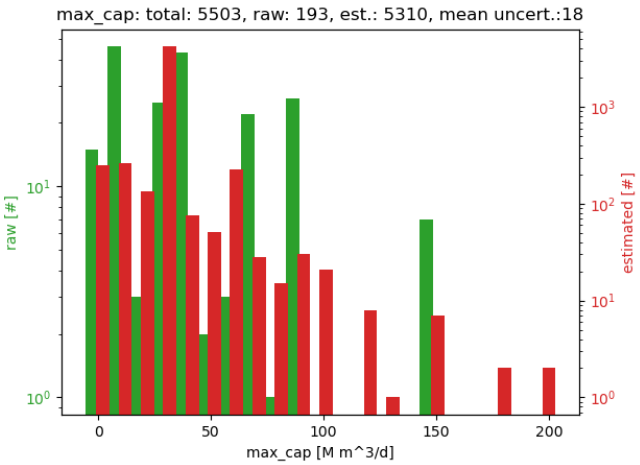


Figure 7.2: Sample plot of the raw and estimated values of the attribute *max_cap_M_m3_per_d* of the component *PipeSegments* on a log Y-axis. Green bars are the raw input values, red bars are the histogram of the estimated values.

  - Number of raw input values
  - The number of generated attribute values
  - The overall mean uncertainty is the last value in the title.
- Below each graph a list of the methods used in generating the missing values is given. Each line is structured as follows:
  - The name of the method
  - In brackets the name(s) of the independent variable
  - The number of attributes that have been generated with these methods
  - In brackets ("mean uncert.") the mean uncertainty of this method for those elements
  - The last line is a summary of the min and maximum raw and estimated values.

The order of the methods listed below the plots does not reflect the order of the application of those methods to generate the missing values. In the automated heuristic attribution generation process, the method with the lowest uncertainties were used before the methods with the higher uncertainties.

With those plots and the additional information in text format below the plots, the user can get an overview of how the missing values were generated, and a summary of their associated uncertainty, hopefully leading to more confidence in the generated data.

7.1.2 Storages

Overall there are 297 *Storages* elements in the final IGGIELGNC-3 data set. The Table 7.3 depicts the most important attributes that are part of the *Storages* elements.

Currently there are no logical/physical heuristics methods implemented for any of the attributes of the component *Storages*. Hence only a single table for the component *Storages* will be presented here, where the raw data will be compared with those values that were generated using the statistical heuristic method.

Table 7.3: List of attributes of *Storages* elements for IGGIELGNC-3 data sets, for the raw and statistically generated values with statistical properties for the most important attributes.

Attribute name	N(R)	N(S)	Ave	Med	P(10)	P(90)	U(S)	Z+(S)
max_cap_store2pipe_M_m3_per_d	188	109	14.3	9.46	2.82	29.0	13.0	4.05
max_cap_pipe2store_M_m3_per_d	178	119	11.3	7.28	2.17	23.8	11.1	3.55
max_cushionGas_M_m3	113	184	868	390	115	1358	1454	3.25
is_H_gas	33	264	0.98	1.00	1.00	1.00	0.50	2.43
max_power_MW	82	215	13.8	9.07	6.00	16.0	21.8	2.71
num_storage_wells	108	189	30.2	19.0	8.00	40.8	36.2	1.70
max_storage_pressure_bar	102	195	132	124	91.2	171	57.2	1.23
max_workingGas_M_m3	198	99	764	304	82.9	1467	626	0.24
min_storage_pressure_bar	81	216	59.9	60.0	45.0	69.4	24.7	0.08

For the seven attributes *max_cap_store2pipe_M_m3_per_d*, *max_cap_pipe2store_M_m3_per_d*, *max_cushionGas_M_m3*, *is_H_gas* and *max_power_MW*, *num_storage_wells* and one can see that the estimated value significantly changed the distribution of the attribute values. By considering the information from the *Storages* supplied through Chapter 10.8, one can see that a very large portion of values were estimated using the median method. Increased data density of the input data set or an attribute-specific non-statistical process might reduce this skewed outcome in future.

For the following two attributes *max_storage_pressure_bar* and *min_storage_pressure_bar*, the **Z+**-score is smaller than 2, indicating that the estimated distribution is similar to the input data distribution. However, a closer look at the methods used show that the method median was the dominant method.

7.1.3 LNGs

Overall there are 32 *LNGs* elements in the final data set. The Table 7.4 depicts all important attributes of the *LNGs* component. Attributes for the component *LNGs* were only given in the data sets INET and GIE, and were explained in previous SciGRID_gas documentations (e.g. [DPM20c]). Here a summary of attribute value distribution for some attributes is given in the Table 7.4.

Table 7.4: List of attributes of *LNGs* elements for the IGGIELGNC-3 data sets, for the raw and statistically generated values, with additional statistical properties for each attribute.

Attribute name	N(R)	N(S)	Ave	Med	P(10)	P(90)	U(S)	Z+(S)
GCV_mean_kWh_per_m3	21	11	11.7	11.6	11.6	11.9	0.21	0.01
max_cap_store2pipe_M_m3_per_d	30	2	25.7	19.5	10.6	47.7	13.4	2.04
max_vessel_size_M_m3	22	10	20.3	128	41.2	156	16.7	0.37
max_workingGas_M_m3	30	2	206	175	49.3	360	64.6	1.94
median_cap_store2pipe_M_m3_per_d	20	12	24.8	19.6	9.63	47.6	3.70	1.46

The **Z+**-score values are larger than 2 for the attributes *max_cap_store2pipe_M_m3_per_d* only. This indicates that the distribution of the raw and the estimated data set are slightly different. One can clearly associate the large difference to the small number of estimated values for the attributes *max_cap_store2pipe_M_m3_per_d*.

7.1.4 BorderPoints

Overall there are 109 *BorderPoints* elements in the final data set. The overall attribute density for elements of type *BorderPoints* is very high, as for almost all attributes, values were supplied.

Table 7.5: List of attributes of *BorderPoints* elements for the IGGIELGNC-3 data sets, with additional statistical properties for each attribute.

Attribute name	N(R)	N(E)	Ave	Med	P(10)	P(90)	U(E)	Z+
GCV_mean_kWh_per_m3	87	22	11.3	11.3	11.1	11.6	0.21	0.22
max_cap_from_to_M_m3_per_d	103	6	26.9	14.7	1.75	64.1	20.0	2.74
max_cap_to_from_M_m3_per_d	100	9	8.94	0.00	0.00	26.6	9.74	4.33

7.1.5 Compressors

Overall there are 248 *Compressors* elements in the final data set. The Table 7.6 depicts the most important attributes that are part of the *Compressors* component. However, even though the number of compressors was increased through the GB data set, the GB data set did not contain any attribute information, such as *max_cap_M_m3_per_d*, *max_power_MW* or *max_pressure_bar*.

Table 7.6: List of attributes of *Compressors* elements for the IGGIELGNC-3 data sets, with additional statistical properties for each attribute.

Attribute name	N(R)	N(E)	Ave	Med	P(10)	P(90)	U(E)	Z+
turbine_power_4_MW	3	245	0.43	0.00	0.00	0.00	0.59	7.28
turbine_fuel_isGas_2	34	214	0.97	1.00	1.00	1.00	0.50	2.97
turbine_fuel_isGas_1	35	213	0.97	1.00	1.00	1.00	0.50	2.96
turbine_power_3_MW	13	235	12.1	12.5	12.5	12.5	4.27	2.91
turbine_power_5_MW	2	246	0.19	0.00	0.00	0.00	0.52	2.71
turbine_power_2_MW	18	230	12.0	11.8	11.8	11.8	4.35	2.04
is_H_gas	245	3	0.98	1.00	1.00	1.00	0.50	2.02
turbine_power_1_MW	19	229	12.0	11.8	11.8	11.8	4.44	1.67
max_cap_M_m3_per_d	18	230	37.3	37.2	22.4	45.6	23.1	1.63
turbine_fuel_isGas_3	22	226	0.99	1.00	1.00	1.00	0.50	1.48
max_power_MW	36	212	40.1	38.3	38.3	38.3	4.67	1.42
max_pressure_bar	17	231	94.6	94.7	94.7	94.7	6.39	0.08
num_turb	37	211	3.00	3.00	3.00	3.00	0.78	0.12

For the attributes *max_cap_M_m3_per_d*, *max_power_MW* and *max_pressure_bar*, a large number of values needed to be estimated, from an input data set of as little as 17 values. For the attributes *max_pressure_bar*, the most missing values were generated using the median of the raw input values. For the attribute *max_cap_M_m3_per_d* the missing values were generated using Lasso Linear regressions with other attribute values, hence resulting in varying attribute values, which are similar in distribution to the input values. For the attribute *max_power_MW*, almost all missing values were derived using the method Lasso implemented with the attribute *num_turb* and *turbine_power_4_MW*.

For most other attributes, the **Z+**-score is around 2 or larger than 2, indicating that the distribution of the estimated values is different to the distribution of the raw input data. Here, it needs to be pointed out that all missing values for the attributes *turbine_fuel_isGas_1* to *turbine_fuel_isGas_6* were set to 1 as a blanket rule, as there were no heuristic capabilities of estimating the missing gas type values. For the attribute of the power of the compressors, most missing values were estimated using the median approach, and for turbine power numbers of 4 and larger a value of 0 was applied for all missing values.

7.1.6 Productions

Overall there are 104 *Productions* elements in the final data set. Information for this component mainly came from the EMAP data set, which contained 103 elements throughout Europe, however not supplying any attribute values, such as capacity or start year. The LKD data set is the only other data set that supplied a further 6 elements for Germany, with some information on gas type and maximum production. Here some information for the IGGIELGNC-3 data set will be presented in Table 7.7.

Table 7.7: List of attributes of *Productions* elements for the IGGIELGNC-3 data sets, with additional statistical properties for each attribute.

Attribute name	N(R)	N(E)	Ave	Med	P(10)	P(90)	U(E)	Z+
is_H_gas	6	96	1.00	1.00	1.00	1.00	0.50	N/A
max_supply_M_m3_per_d	5	97	1252	1230	1230	1230	855	0.92

The automated heuristic process achieved lowest fitting uncertainty by using the median of the input data. This was to be expected, as there were only 5 elements with a value for the attribute *max_supply_M_m3_per_d*. Hence all 101 estimated values for *max_supply_M_m3_per_d* have the same value of 1319 $\mathrm{Mm^3 d^{-1}}$, with a large uncertainty of 916 $\mathrm{Mm^3 d^{-1}}$. The uncertainty is very large in respect to the absolute value, which is understandable due to the small

number of training values. However, there is no other information in respect of production for those gas production sites.

7.1.7 PowerPlants

Overall there are 310 NUTS-3 *PowerPlants* elements in the final data set. Information for this component originated from the INET and the CONS data sets. Here information for the IGGIELGNC-3 data set will be presented in Table 7.8.

Table 7.8: List of attributes of *PowerPlants* elements for the IGGIELGNC-3 data sets, with additional statistical properties for each attribute.

Attribute name	N(R)	N(E)	Ave	Med	P(10)	P(90)	U(E)	Z+
capacity_E_MW	310	0	582	231	26.23	1562	N/A	N/A
capacity_TH_MW	21	8	512	377	43.8	1219	0.00	0.285
is_H_gas	0	310	1.00	1.00	1.00	1.00	0.50	N/A

This data source contained information for all the elements for the attribute *capacity_E_MW* and *is_gas_fuel1*. As no attribute values needed to be estimated for *capacity_E_MW*, the **Z+**-score was not derived. A further attribute was included for interest only: *capacity_TH_MW* (thermal energy produced for the heat network/processes). Only a small number of power plants came with a value, hence for more than 90 % of power plants, this value was estimated. This does not indicate that all of those power plants are connected to a heat grid. No information was given from the original data sets in that respect. Here, the thermal power estimated is highly correlated to the electric generated installed power *capacity_E_MW*. In addition, it was assumed that all 310 power stations were using the high calorific gas, hence as no raw input data existed, the attribute value *is_H_gas* was set to one.

7.1.8 Consumers

Overall there are 1367 NUTS-3 *Consumers* elements in the final data set. Information for this component originated from the INET and the CONS data sets. Here information for the IGGIELGNC-3 data set will be presented in Table 7.9.

Table 7.9: List of attributes of *Consumers* elements for the IGGIELGNC-3 data sets, with additional statistical properties for each attribute.

Attribute name	N(R)	N(E)	Ave	Med	P(10)	P(90)	U(E)	Z+
max_demand_M_m3_per_d	1357	0	1.45	0.97	0.26	3.08	0.05	N/A
mean_demand_M_m3_per_d	1357	0	0.54	0.38	0.11	1.14	0.05	N/A
median_demand_M_m3_per_d	1357	0	0.49	0.35	0.10	0.99	0.05	N/A
min_demand_M_m3_per_d	1357	0	0.23	0.16	0.09	0.47	0.05	N/A

This data source contained information for all the elements of all the attributes, as they were generated outside of this project. As no attribute values needed to be estimated for those attributes, the **Z+**-scores were not derived.

7.1.9 Nodes

Overall there are 5009 *Nodes* elements in the final data set. Each original data set contributed some or many *Nodes* elements to the final data set. The nature of the *Nodes* elements is to supply the topological information only. Any latitude and longitude values were derived from the original data set, and any height information was derived using the BING or opentopodata.org web API.

7.1.10 Summary

The IGGIELGNC-3 data set is a further data set created as part of the SciGRID_gas project. For each component the number of elements, and the attribute data density was presented. It was pointed out that for some attributes, other methods of missing value generation need to be found, as the distribution of the estimated attribute values was significantly different to the value distribution of the raw input data. Here further input data and attribute-specific heuristic methods should help in determining "better" missing values. On the other hand, other missing attribute values could easily be generated with the implemented automated attribute generation process.

7.1.11 Resulting map of data set

Below a spatial presentation of the final IGGIELGNC-3 data set is given in Figure 7.3, resulting in a network of 236,928 km in length. In addition, the number of elements for each component is listed in Table 7.10.

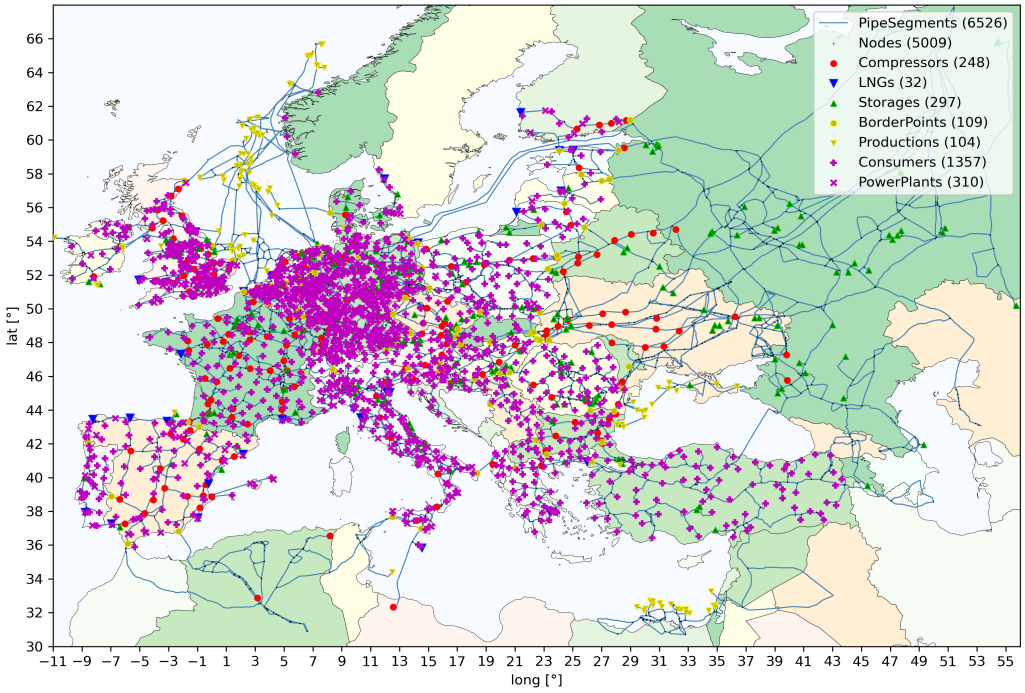


Figure 7.3: Map of the final IGGIELGNC-3 data set.

Table 7.10: List of components with number of elements of the final merged and filled IGGIELGNC-3 network data set.

Component name	Number of elements
BorderPoints	109
Compressors	248
Consumers	1357
LNGs	32
Nodes	5009
PipeSegments	6526
PowerPlants	310
Productions	104
Storages	297

CHAPTER

EIGHT

SENSITIVITY ANALSYS OF HEURISTIC METHODS

In the previous sections of the documentation the different data sets were introduced, methods described to merge those data sets, and futher explained how missing values were generated through the logical and statistical heuristic data generation processes. Even though uncertainty values are part of the output data set (see Chapter 2.1), where for each attribute value an uncertainty was specified that was caused by the heuristic process, it would be of great interest to compare the different generated value in case that one selects only a subset of heuristic methods. Hence, this section describes the sensitivity of the different heuristic methods that were implemented to generate the missing attribute values.

8.1 Description of approach

Chapter Chapter 5 described the different heuristics implemented to generate missing attribute values. As previously described there were two different method groups of heuristics, a physical/logical based one and a pure statistical based method group. Here the pure statistical based method group options will be investigated, so that the user can be assured, that the methods used are the most favourable one. Here one of the last published SciGRID_gas data sets will be used as an example data set to demonstrate the sensitivity. it is the IGGIELGN data set ([DPM20c]), where here the attribute *min_storage_pressure_bar* will be investigated from the component *Storages*. At the process step where missing values were estimated, the IGGIELGN data set contained 300 elements of type *Storages*. Of those 300, 83 supplied a value for the attribute *min_storage_pressure_bar*. As was described in Chapter 5.2.2, different statistical methods were coded, such as **Linear-Lasso**, **Mean** and **Median**. As was explained further in Chapter 5.4, the method with the smallest mean absolute error was selected, followed by the method with the next smallest mean absolute error, and this process was repeated until all missing values were estimated. As mentioned, here the method selection was carried out in respect of the error estimate **mean absolute error**, where Figure 5.15 and Figure 5.16 depict the internal meta data for the method selection process. In respect of sensitivity of the model selected, those tables give the user the information needed. Different statistical estimation models (column D) with different independent variable inputs (column F) were used as options, and the corresponding *mean absolute error* was simulated and depicted in column L. Hence, the user can see, if only using the second best option (here method *Lasso* and independent variable input *Pipe_max_cap_M_m3_per_d* would have lead a slightly higher **MAE(E)** of 4.85, when compared to the best option of using the *Lasso* method with the independent variable *median_cap_store2pipe_M_m3_per_d*, which resulted in a **MAE(E)** of 4.75 $Mm^3 1/d$.

Here it will be demonstrated, by how much the estimated values differ to the raw input value when a sub-optimal model with independent variable were selected, to estimate all unknown attributes. The above-mentioned attribute *min_storage_pressure_bar* from the component *Storages* as part of the IGGIELGN data set will be used.

Table 8.1 depicts different simple model methods: **Max**, **Min**, **Mean** and **Median**, where the raw values for the attribute *min_storage_pressure_bar* were the input variable. For testing of the model, the **mean absolute error** (**MAE(E)**) and the **Z**-score are also presented as defined in Chapter 5.2. The raw data set contained 84 values for the 301 *Storages* elements. As can be seen from Table 8.1, by estimating the 217 missing values and using the **Max** model, then the overall **MAE(E)** was large with a value of 88.16 bar, and a **Z+**-score of 27. By using the minimum value of the raw input data set to estimate all missing values, the **MAE(E)** reduced to 48.62 bar and the **Z+**-score also

reduced to almost 15. However, for both model methods, the **Z+**-score is indicating, that the data distribution of the raw input data and the distribution of the estimated values is totally different, however using the **Min** over the **Max** would have improved the estimation of the missing values (represented by the smaller **MAE(E)** value. A further model method was introduced, the **Mean** model method. Here all missing values are filled with the **Mean** of the raw input values. Here a large reduction in corresponding **MAE(E)** can be seen, in addition to a **Z+**-score of zero. Here, one should ignore the **Z+**-score of zero, as outlines in Chapter 10.9.6. But the lower **MAE(E)** value shows that the **Mean** model performs better than the **Max** or the **Min** model. In addition, the **Median** model method was also introduced, resulting in very similar values of **MAE(E)** and **Z+**-score values when compared with the **Mean** model method.

Model method	independent variables	**MAE(E)**	**Z+**-score
Max	*min_storage_pressure_bar*	88.2	27.0
Min	*min_storage_pressure_bar*	48.6	14.9
Mean	*min_storage_pressure_bar*	24.8	0
Median	*min_storage_pressure_bar*	24.7	0.42

Next to the above introduced model methods **Max**, **Min**, **Mean** and **Median**, the following other statistical model methods were introduced: **Lasso**, **Squared**, **OneOver** and **Logarithmic** as described in Chapter 5.2.2. To investigate how sensitive the output is in respect to new models introduced, the following pathway has been carried out, with results presented in Table 8.1. This table contains two more columns:

- **N(M)**: This column contains the number of missing values, that were generated by the **Median** model method.
- **N(!=M)**: This column contains the number of missing values, that were generated by any other model method, other than the **Median** model method.

First the missing values were estimated using the model method **Median** only (first line in Table 8.1) with results of the resulting **MAE(E)** and **Z+**-score of 24.74 bar and 0.41 respectively. In addition, the uncertainty of this individual model method is given through the column **MAE(MM)**, and as the **Median** model method was the only one applied to the missing values, the values for **MAE(MM)** and **MAE(E)** are identical.

In the next step, the model method **Lasso** with input variables *max_workingGas_M_m3* and *Pipe_max_cap_M_m3_per_d* were added. Here this **Lasso** model method was execute first, resulting in the estimation of only two values (see column **N(!=M)** in the second line in Table 8.1), whereas all other missing values were generated with the **Median** model method. The overall **MAE(E)** reduced slightly to 24.57 bar, whereas for two *Storages* elements the uncertainty of the estimated attribute is 5.36 bar, given through column **MAE(MM)**.

In the next step, the model method **Lasso** with input variables *Pipe_max_cap_M_m3_per_d* was added and executed first, to estimate missing values (see third line in Table 8.1). Then in a second step the previous model method **Lasso** with input variables *max_workingGas_M_m3* and *Pipe_max_cap_M_m3_per_d* was executed (second line in Table 8.1), and this was followed by applying the **Median** model method (first line in Table 8.1). As can be seen from the entries in the first line in Table 8.1), the **MAE(E)** slightly decreased, and as well, the **Z+**-scored slightly reduced as well. Just for clarity, the values of **N(M)**, **N(!=M)**, **MAE(E)**, and **Z+**-score from Table 8.1 are the overall scored, when applying the model methods as described above. Whereas the column **MAE(MM)** is the uncertainty of the model method and independent variable combination of corresponding line only.

In the next steps, more and more model methods are being added. As one can see, the value of **N(!=M)** stagnates at 17, meaning even though more model methods were added, the number of missing values estimated with the non-**Median** model method did not increase. This is due to the sparse underlying input data set.

By adding more model methods, one can see that the **MAE(E)** decreases slightly, hence introducing further model methods, such as **Lasso** for attributes *Pipe_max_cap_M_m3_per_d* and *Pipe_diameter_mm* reduced the overall uncertainty of the estimated values (**MAE(E)**).

Model method	independent variables	N(M)	N(!=M)	MAE(MM)	MAE(E)	Z+-score
Median	*min_storage_pressure_bar*	217	0	24.74	24.74	0.41
Lasso	*max_workingGas_M_m3*, *Pipe_max_cap_M_m3_per_d*	215	2	5.36	24.57	0.46
Lasso	*Pipe_max_cap_M_m3_per_d*	211	6	5.36	24.21	0.29
Squared	*max_cap_store2pipe_M_m3_per_d*, *Pipe_max_pressure_bar*	200	17	5.03	23.21	0.71
Lasso	*max_workingGas_M_m3*, *Pipe_max_pressure_bar*	200	17	5.02	23.21	0.76
Lasso	*max_cap_store2pipe_M_m3_per_d*, *Pipe_max_pressure_bar*	200	17	4.61	23.18	0.41
Lasso	*Pipe_max_cap_M_m3_per_d*, *Pipe_pipe_class_EMap*	200	17	4.12	23.15	0.48
Squared	*Pipe_max_cap_M_m3_per_d*, *Pipe_pipe_class_EMap*	200	17	3.97	23.15	0.37
Squared	*Pipe_max_cap_M_m3_per_d*, *Pipe_diameter_mm*	200	17	3.19	23.13	0.41
Lasso	*Pipe_max_cap_M_m3_per_d*, *Pipe_diameter_mm*	200	17	3.11	23.13	0.51

With this approach described above and implemented in the overall SciGRID_gas attribute generation process, it is tried to demonstrate, that by adding more model methods (combination of regression methods and different independent variables) the uncertainty of the estimated values was reduced. Best reduction was achieved by implementing a pathway where model methods were selected with smallest **MAE(E)**.

In this chapter it was demonstrated, how sensitive the estimation of missing values is in respect to selected model method. A model method is a combination of one of the statistical regression methods (such as liner **Lasso** or **Logarithmic**) and a set of independent input variables. It was demonstrated, that by expanding the options of model methods, the error associated with estimated missing values can be reduced. The larger the pool of model methods, the smaller the overall error in the estimated values. Here in the SciGRID_gas project, a large set of statistical regression methods has been implemented, and it was also tried to increase the number of possible independent variables, by moving attribute values from connecting components as was explained in Chapter 5.2.

CHAPTER

NINE

CONCLUSION

This document describes one of the data sets that are generated as part of the SciGRID_gas project. It starts off with the introduction of the SciGRID_gas project, such as funding, duration and goals. In a subsequent chapter the data structure within the SciGRID_gas project is described, such as components, elements, attributes and attribute values. The third chapter introduced all the different individual data sources: INET, GIE, GSE, IGU, EMap, LKD, GB, NO and CONS data sets. In the next chapter, tools for merging elements are introduced. This is followed by a chapter describing the heuristic generation of any missing attribute value. The final chapter describes briefly the final data set, with its 6526 pipes and more than 240 compressors, where all elements have been connected to a single network, and where all missing attribute values have been estimated using heuristic processes. The final data set is termed "IGGIELGNC-3" data set and spans 236,928 km of transmission pipes over Europe.

CHAPTER

TEN

APPENDIX

10.1 Glossary

Dataset abbreviations can be found in Table 10.1.

Table 10.1: Dataset abbreviations

Name	Abbreviation	Description
Raw InternetDaten data set	INET	Label/name for the raw InternetDaten data set
Raw Gas Infrastructure Europe data set	GIE	Label/name for the raw Gas Infrastructure Europe data set
Raw Gas Storage Europe data set	GSE	Label/name of the raw Gas Storage Europe data set
Raw Norwegian data set	NO	Label/name for the raw Norwegian data set
Raw Long-term planning and short-term optimization data set	LKD	Label/name for the raw Long-term planning and short-term optimization data set
Raw International Gas Union data set	IGU	Label/name for the raw International Gas Union data set
Raw EntsoG-Map data set	EMAP	Label/name for the raw EntsoG-Map data set
Raw consumer data set	CONS	Label/name for the raw natural gas consumer data set
Merged and filled IGG data set	IGG	Filled data sets, for which the **I**NET, **G**IE and **G**SE data sets were merged
Merged and filled IGGI data set	IGGI	Filled data sets, for which the **I**NET, **G**IE, **G**SE and **I**GU data sets were merged
Merged and filled IGGIN data set	IGGIN	Filled data sets, for which the **I**NET, **G**IE, **G**SE, **I**GU and the **N**O data sets were merged
Merged and filled IGGINL data set	IGGINL	Filled data sets, for which the **I**NET, **G**IE, **G**SE, **I**GU, **N**O and the **L**KD data sets were merged
Merged and filled IGGIELGN data set	IGGIELGN	Filled data sets, for which the **I**NET, **G**IE, **G**SE, **I**GU, **E**MAP, **L**KD, **G**B, and the **N**O data sets were merged
Merged and filled IGGIELGNC-3 data set	IGGIELGNC-3	Filled data sets, for which the **I**NET, **G**IE, **G**SE, **I**GU, **E**MAP, **L**KD, **G**B, and the **N**O data sets were merged, where the CONS data was supplied on a NUTS3 level
Merged and filled IGGIELGNC-2 data set	IGGIELGNC-2	Filled data sets, for which the **I**NET, **G**IE, **G**SE, **I**GU, **E**MAP, **L**KD, **G**B, and the **N**O data sets were merged, where the CONS data was supplied on a NUTS2 level
Merged and filled IGGIELGNC-1 data set	IGGIELGNC-1	Filled data sets, for which the **I**NET, **G**IE, **G**SE, **I**GU, **E**MAP, **L**KD, **G**B, and the **N**O data sets were merged, where the CONS data was supplied on a NUTS1 level

The glossary terms can be found in Table 10.2.

Table 10.2: Glossary

Name	Abbreviation	Description
component		A gas network consists of different components, such as: pipelines, compressors, LNG terminals, storages, entry points and production sites
element		Elements are instances of components. Hence, "10 compressor elements" refers to a data set that contains information for 10 compressor stations
attribute		Gas facilities, such as pipelines or compressors, can be described with a large set of parameters, such as pipeline diameter, or compressor capacity. Those parameters are referred to as attributes
facility		General term used for a gas appliance, such as a single compressor station, or a single LNG terminal
PipeLine		A gas pipeline entity, which has one start and one end point, however, can run via many nodes
PipeSegment		A gas pipeline that has only one start and one end point, but no nodes in-between
LNG	LNG	Liquefied natural gas
CNG	CNG	Compressed natural gas
flow duration curve	FDC	It is the cumulative frequency curve that shows the percentage of time specified flow where equal or exceeded during a given period. The temporal information, when certain events occur, is lost
Energiewende		German term for the change in using primary energies, the move away from coal to renewable energies, such as wind or solar
gas component data set		Raw input data, associated with components of the gas transmission grid
gas network data set		Output data, a coherent network of gas transmission components
OSM	OSM	Data that is available from openstreetmap.org
non-OSM	Non-OSM	Data that is not part of the OSM data set
gas type		There are two types of gas: High (H) and Low (L) calorific gas
mean absolute error	MAE	mean difference between input values and estimated values
data density		The ratio of the number of usable (not missing) attribute values over number elements of the component, in units of [%]
Transmission System Operator	TSO	An entity entrusted with the transportation of natural gas/electricity, as defined by the European Union
gas transmission network		This describes the physical gas transmission grid, however, it excludes any facilities/components that would be part of a distribution network and their facilities
gas component data set		The term "gas component data set" is used for raw data sets of gas network facilities. However, not all elements (e.g. compressors) need to be connected to pipelines, where the emphasis is on the term **component**
gas network data set		A "gas component data set" can be converted into a "gas network data set", by connecting all non-pipeline elements to nodes and all nodes are connected to pipelines. Hence, the emphasis here is on the term **network**
Nomenclature des unités territoriales statistiques	NUTS	Geographical system dividing the Eurpean Union into regions of similar size in respect of number of inhabitatnts

10.2 Unit conversions

Table 10.3: Unit conversions

From Unit	To Unit	MultiVal
LNG Mt	LNG Mm^3	2.47
gas $\mathrm{tm}^3\mathrm{h}^{-1}$	gas $\mathrm{Mm}^3\mathrm{d}^{-1}$	24/1000
LNG Mm^3	gas Mm^3	584
LNG t	gas Mm^3	1442.48
GWh (H)	gas Mm^3	0.0879757777
GWh (L)	gas Mm^3	0.1023541453

For some elements of some components, the calorific value was given through the references. Hence during the conversion process from GWh to M m^3, the elements calorific value was used, however, whereever the element specific calorific value was not known, the default values from Table 10.3 was used in dependence of the gas type of the element. If no gas type was known, then high calorific gas is assumed.

10.3 Attribute *exact*

Each element of type *Nodes* has an attribute *exact*. With this, the SciGRID_gas project is trying to let the user know, how well the actual location of the *Nodes* elements are known. The actual location (latitude-longitude pair) can be spot on (verifiable through satellite imagery) or can be unknown by 10's or 100's of km, where city names or country names are known only. Here the attribute value for *exact* is being given, ranging from "1" to "5" as listed in Table 10.4 below.

Table 10.4: Unit conversions

Exact value	Description	Uncertainty [km]
1	The exact location of this node is known, as one was able to verify the facility through satellite data.	0
2	Here the lat/long is not known exactly. However, one assumes that the location is within a small region (e.g. Krummhörn). Hence, not being much larger than 10 km	10
3	Here so little is known about the exact location, and one only knows that the location is within a large region (e.g. Hamburg). Hence, the actual location could be out by 10 km or more but less than 100 km	100
4	Here so little is known about the exact location, and one only knows that the location is within a state (e.g. Niedersachsen). Hence, the actual location could be out by 100 km or more but less than 1000 km	1000
5	Here so little is known about the exact location, and one only knows that the location is within a country (e.g. Ukraine). Hence, the actual location could be out by 1000 km or more.	> 1000

10.4 References for INET data set

Below a list of those sources used to generate the INET data set.

- Christoph Edler, Bachelorarbeit PR 370005, Technische Universität Wien, "Das österreichische Gasnetz", Juli 2013.
- http://belarus-tr.gazprom.ru/
- http://corporate.vattenfall.com/about-vattenfall/operations/market-transparency/gas-storage/
- http://digitalnewsservice.net/clients/net4gas-nimmt-neue-hochdruck-gas-pipeline-gazelle-in-betrieb/
- http://en.gaz-system.pl//
- http://gaslager.energinet.dk/EN/Pages/default.aspx
- http://interfaxenergy.com/article/19138/lng-better-than-norway-pipeline-ex-polish-pm
- http://ir.gasplus.it/home/show.php?menu=00002
- http://italgasstorage.it/eng/progetto.html
- http://media.edfenergy.com/Misc/AboutUs.aspx
- http://mmbf.hu/en/company/gas_storage
- http://mndgsgermany.com/
- http://mysolar.cat.com/cda/files/870985/7/Solar_Turbines_5000_Gas_Compressor_News_Rel_Sent.pdf
- http://sse.com/whatwedo/ourprojectsandassets/thermal/Aldbrough/
- http://sse.com/whatwedo/wholesale/gasstorage/
- http://www.bayernugs.de/4-1-Home.html
- http://www.bendisenergy.com.tr/tr/Projelerimiz/24-toren-dogalgaz-depolama-ve-madencilik-as
- http://www.berliner-erdgasspeicher.de/en/Pages/default.aspx
- http://www.botas.gov.tr/
- http://www.bulgartransgaz.bg/en/pages/transstorge-110.html
- http://www.caythorpegasstorage.com/caythorpe/
- http://www.centrica-sl.co.uk/
- http://www.ceskaplynarenska.cz/en/ultimate-speed-underground-gas-storage
- http://www.dea-speicher.de/en
- http://www.depomures.ro/
- http://www.edisonstoccaggio.it/en
- http://www.ekb-storage.de/de/home/
- http://www.emplpipeline.com/en/the-gas-pipeline/
- http://www.enagas.es/enagas/en/Transporte_de_gas/Almacenamientos_Subterraneos
- http://www.energystock.com/
- http://www.ewe-gasspeicher.de/english/index.php
- http://www.fluxys.com/belgium/en/Services/Storage/Storage

- http://www.friedrich-vorwerk.de/de/aktuell/projekte/neubau-nowal-dn-1000.htmleilung/news/bau-der-nord-west-anbindungsleitung-nowal-startet-anfang-maerz/
- http://www.gasnaturalfenosa.com/en/activities/lines+of+business/1285338591925/supply+and+transportation+of+gas.html
- http://www.gasspeicher-hannover.de/startseite.html
- http://www.gasstorage.cz/en/operation-information/available-firm-capacity/
- http://www.gasstoragebergermeer.com/gas-storage-bergermeer-2/
- http://www.gastrade.gr/en/the-company/the-project.aspx
- http://www.gas-union-storage.de/
- http://www.gatewaystorage.co.uk/
- http://www.gazprom.com/about/production/underground-storage/
- http://www.geogastock.it/ITA/Home.asp
- http://www.grtgaz.com/en/major-projects/beynes-compressor-station/presentation/news/compressor-station-at-the-beynes-site.html
- http://www.grtgaz.com/fileadmin/plaquettes/en/2017/Essentiel-plaquette-institutionnelle-EN-2017.pdf
- http://www.gsa-services.ru/
- http://www.halite-energy.co.uk/our-project/project-overview/
- http://www.hradf.com/en/portfolio/south-kavala-natural-gas-storage
- http://www.humblyenergy.co.uk/about-us#useful-information
- http://www.islandmageestorage.com/
- http://www.kge-gasspeichergesellschaft.de/
- http://www.kgsp.co.uk/
- http://www.kingstreetenergy.com/
- http://www.kinsaleenergy.ie/gas-storage.html
- http://www.le.lt/index.php/projects-in-progress/syderiai-underground-gas-storage/535
- http://www.lg.lv/index.php?id=3376&lang=eng
- http://www.magyarfoldgaztarolo.hu/en/Lapok/default.aspx
- http://www.mnd.eu/en/2014-11-26-12-13-02/mnd-group-companies
- http://www.nafta.sk/en/about-gas-storage
- http://www.nam.nl/en/about-nam/facts-and-figures.html
- http://www.omv.com/portal/01/com/gas/storage
- http://www.petroceltic.com/operations/bulgaria.aspx
- http://www.pgnig.pl/reports/annualreport2012/en/ar-obrot-magazynowanie-2.html
- http://www.pozagas.sk/en/?PHPSESSID=8d83cc7890abb7980f6793cf56633a72
- http://www.psp.hr/home
- http://www.rag-energy-storage.at/en.html
- http://www.romgaz.ro/en/content/ugs-n-366-new-gas-storage-facility-romania

- http://www.romgaz.ro/en/content/ugs-n-371-sarmasel-storage-facility-upgrading
- http://www.romgaz.ro/en/inmagazinare
- http://www.rwe.com/web/cms/de/37110/rwe/presse-news/pressemitteilungen/?pmid=4005467
- http://www.rwe.com/web/cms/en/531750/rwe-gasspeicher/
- http://www.scottishpower.com/pages/hatfield_moor_gas_storage_facility.asp
- http://www.snam.it/en/transportation/Thermal_Year_Archive/Thermal_Year_2013_2014/Info-to-users/index.html
- http://www.sppstorage.cz/#
- http://www.srbijagas.com/naslovna.1.html
- http://www.stogit.it/en/index.html
- http://www.storengy.com/countries/deutschland/en/products-services.html
- http://www.storengy.com/countries/france/en/
- http://www.storengy.com/countries/unitedkingdom/en/oursites.html
- http://www.swissgas.ch/en/3_2.php
- http://www.taqaglobal.com/our-regions/netherlands/gas-storage/peak-gas-installation/overview?sc_lang=en
- http://www.terranets-bw.de/en/gas-transmission/we-transport-your-natural-gas/
- http://www.thueringerenergie.de/Unternehmen/Ueber_uns/Geschaeftsfelder/Speicher.aspx
- http://www.tigf.fr/en/what-we-can-offer/storage.html
- http://www.tpao.gov.tr/eng/?tp=m&id=31
- http://www.tpao.gov.tr/eng/?tp=m&id=84
- http://www.trianel-gasspeicher.com/
- http://www.ugs-katharina.de/en/unternehmen.html
- http://www.vng-gasspeicher.de/content/en/Speicher/index.html
- https://de.wikipedia.org/wiki/Baumgarten_an_der_March
- https://de.wikipedia.org/wiki/Erdgasleitung_Jamal%E2%80%93Europa
- https://de.wikipedia.org/wiki/Hungaria-Austria-Gasleitung
- https://de.wikipedia.org/wiki/MIDAL
- https://de.wikipedia.org/wiki/Mittel-Europ%C3%A4ische_Gasleitung
- https://de.wikipedia.org/wiki/NEL_(Pipeline)
- https://de.wikipedia.org/wiki/Norddeutsche_Erdgas-Transversale
- https://de.wikipedia.org/wiki/OPAL_(Pipeline)
- https://de.wikipedia.org/wiki/Penta-West
- https://de.wikipedia.org/wiki/Rehden-Hamburg-Gasleitung
- https://de.wikipedia.org/wiki/STEGAL
- https://de.wikipedia.org/wiki/Trans_Austria_Gasleitung
- https://de.wikipedia.org/wiki/Trans-Adria-Pipeline

- https://de.wikipedia.org/wiki/Trans-Europa-Naturgas-Pipeline
- https://de.wikipedia.org/wiki/Transgas-Pipeline
- https://de.wikipedia.org/wiki/WEDAL
- https://de.wikipedia.org/wiki/West-Austria-Gasleitung
- https://en.wikipedia.org/wiki/Arad%E2%80%93Szeged_pipeline
- https://en.wikipedia.org/wiki/BBL_Pipeline
- https://en.wikipedia.org/wiki/BRUA_Pipeline
- https://en.wikipedia.org/wiki/Europipe_II
- https://en.wikipedia.org/wiki/Giurgiu%E2%80%93Ruse_pipeline
- https://en.wikipedia.org/wiki/MEGAL_pipeline
- https://en.wikipedia.org/wiki/National_Transmission_System
- https://en.wikipedia.org/wiki/NEL_pipeline
- https://en.wikipedia.org/wiki/Netra
- https://en.wikipedia.org/wiki/OPAL_pipeline
- https://en.wikipedia.org/wiki/Rehden%E2%80%93Hamburg_gas_pipeline
- https://en.wikipedia.org/wiki/Scotland-Northern_Ireland_pipeline,E127
- https://en.wikipedia.org/wiki/South_Wales_Gas_Pipeline
- https://en.wikipedia.org/wiki/STEGAL
- https://en.wikipedia.org/wiki/Trans_Adriatic_Pipeline
- https://en.wikipedia.org/wiki/Trans_Austria_Gas_Pipeline
- https://en.wikipedia.org/wiki/Transitgas_Pipeline
- https://en.wikipedia.org/wiki/V%C3%A1rosf%C3%B6ld%E2%80%93Slobodnica_pipeline
- https://en.wikipedia.org/wiki/Yamal%E2%80%93Europe_pipeline
- https://globallnghub.com/wp-content/uploads/2018/09/King.pdf
- https://nauticor.de/lng-terminal-nynaeshamn
- https://news.err.ee/610905/vopak-to-build-initially-4-000-cubic-meter-lng-terminal-at-muuga
- https://theodora.com/pipelines/france_and_belgium_pipelines.html
- https://theodora.com/pipelines/france_and_belgium_pipelines.html, E21
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E10
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E106
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E11
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E112
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E113
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E13
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E14
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E16_1

- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E16_2
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E17_Skikda
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E22_Arzew_1
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E22_Arzew_2
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E48
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E64
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E65
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E78
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E80_1
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E80_2
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E81
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E85
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E86
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E87
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E88
- https://theodora.com/pipelines/france_and_belgium_pipelines.html,E91
- https://transparency.entsog.eu/
- https://web.archive.org/web/20070808033932/
- https://web.archive.org/web/20070808033932/http://www.swissgas.ch/en/3_2.php
- https://www.astora.de/storage-locations/haidach-storage-facility.html?L=1
- https://www.astora.de/storage-locations/jemgum-storage-facility.html?L=1
- https://www.astora.de/storage-locations/jemgum-storage-facility.html?L=2
- https://www.astora.de/storage-locations/rehden-storage-facility.html?L=1
- https://www.baltic-pipe.eu/de/das-projekt/
- https://www.enagas.es/stfls/ENAGAS/Transporte%20de%20Gas/Documentos/CAT_English.pdf
- https://www.enbw.com/unternehmen/konzern/geschaeftsfelder/speicher/gasspeicher/zahlen-daten-fakten.html
- https://www.eneco.nl/over-ons/projecten/gasspeicher/voorraadniveau
- https://www.energiefachmagazin.de/Branchen-News/VNG-Gasspeicher-GmbH-legt-Erdgasspeicher-Buchholz-still
- https://www.enovos.de/industrie/ueber-uns/erdgasspeicher
- https://www.eugal.de/eugal-pipeline/
- https://www.europipe.com/de/referenzen/referenzprojekte/
- https://www.eustream.sk/en_transmission-system/en_transmission-system
- https://www.fluxys.com/belgium/en/about%20fluxys/infrastructure/network/network
- https://www.fluxys.com/nel/en/NELSystemInfo/AboutNEL
- https://www.fluxys.com/tenp/de

- https://www.fnb-gas.de/media/FNB_GAS_Projekte_2014_02_17_anlage_6_nep-gas-2014_projekt-steckbriefe.pdf
- https://www.gascade.de/Anlandestation_Greifswald_Lubmin_2016.pdf
- https://www.gascade.de/Compressor_station_Eischleben_2016.pdf
- https://www.gascade.de/Compressor_station_Olbernhau_2016.pdf
- https://www.gascade.de/Compressor_station_Rueckersdorf_2016.pdf
- https://www.gascade.de/en/our-network/compressor-stations/radeland/
- https://www.gascade.de/en/our-network/our-pipelines/jagal/
- https://www.gascade.de/netzinformationen/unser-leitungsnetz/midal/
- https://www.gascade.de/netzinformationen/unser-leitungsnetz/stegal/
- https://www.gascade.de/netzinformationen/unser-leitungsnetz/wedal/
- https://www.gascade.de/presse/presseinformationen/pressemitt
- https://www.gascade.de/Verdichterstation_Bunde_2016.pdf
- https://www.gascade.de/Verdichterstation_Lippe_72dpi060614.pdf
- https://www.gascade.de/Verdichterstation_Mallnow_2016.pdf
- https://www.gascade.de/Verdichterstation_Reckrod_2016.pdf
- https://www.gascade.de/Verdichterstation_Rehden_2016.pdf
- https://www.gasinfocus.com/en/indicator/existing-and-planned-lng-terminals/
- https://www.gazprom.com/f/posts/86/569604/portovaya_eng.pdf
- https://www.grtgaz.com: Plan_decennal_2017-2026.pdf
- https://www.habau.at/de/projekte/erdgasleitung-wag-expansion-3
- https://www.habau.at/de/projekte/erdgasleitung-wag-plus-600-phase-1
- https://www.hydrocarbons-technology.com/projects/swedegas-lng-facility-port-gothenburg/
- https://www.ign.ren.pt/en/armazenamento-subterraneo3
- https://www.ign.ren.pt/en/armazenamento-subterraneo4
- https://www.innogy-gasstorage.cz/en/index/
- https://www.innogy-gasstorage.cz/en/stramberk/
- https://www.lngworldnews.com/tallinna-sadam-alexela-to-work-on-paldiski-lng-terminal/
- https://www.mdr.de/nachrichten/politik/regional/baubeginn-erdgastrasse-eugal-umstritten-100.html
- https://www.n-ergie.de/geschaeftskunden/produkte/erdgas/erdgasspeicher.html
- https://www.ontras.com/fileadmin/user_upload/Dokumente_Download/Publikationen/ONTRAS_Netzpufferanlage_Burggraf_Bernsdorf.pdf
- https://www.opal-gastransport.de/Compressor_Station_Radeland_72dpi.pdf
- https://www.opal-gastransport.de/netzinformationen/ostsee-pipeline-anbindungsleitung/
- https://www.opal-gastransport.de/Verdichterstation_Radeland_2016.pdf
- https://www.open-grid-europe.com
- https://www.osm.pgnig.pl/en

- https://www.pipelinesystems.com/: "NETRA compressor station Wardenburg"
- https://www.PLE
- https://www.rnf.de/hintergrund-die-erm-gasleitung-von-ludwigshafen-nach-karlsruhe-58182/
- https://www.shz.de/lokales/landeszeitung/wissenschaftler-unter-zeitdruck-id5845016.html
- https://www.sourcewatch.org/index.php/Delimara_Malta_LNG_Terminal
- https://www.sourcewatch.org/index.php/Lithuania-Latvia_Interconnection_Gas_Pipeline
- https://www.storengy.com/countries/france/en/nos-sites/beynes.html
- https://www.storengy.com/countries/france/en/nos-sites/cere-la-ronde.html
- https://www.storengy.com/countries/france/en/nos-sites/manosque.html
- https://www.storengy.com/countries/france/en/our-sites/saint-clair-sur-epte.html
- https://www.streicher.de
- https://www.swedegas.com/Our_services/services/Storage
- https://www.uniper-energy-storage.com/cps/rde/xchg/ust/hs.xsl/3252.htm?rdeLocaleAttr=en
- https://www.uniper-energy-storage.com/cps/rde/xchg/ust/hs.xsl/3437.htm?rdeLocaleAttr=en
- https://www.vng-gasspeicher.de/storage_locations
- https://www.wesernetz.de/netznutzung/bremen/gasnetz-speicheranlagen.php
- https://www.wingas.com/storage-uk-ltd/home.html
- https://www.www.gascade.de/Verdichterstation_Weisweiler_201609.pdf
- NWZonline.de, 22-04-2010: "ExxonMobil gibt kräftig Gas" by Tanja Mikulski
- Porzerleben.de: 6-sep-2011, "Open Grid Europe investiert in europäischen Netzverbund"
- Presseinfo Bayernets, WinGas, 19-Sep-2008

10.5 Location name alterations

Location names should be changed into the 26 letters used in the English language.

For names from the individual countries please follow the suggested approach:

- Germany/Austria: *Umlaute* to be replaced with the letter followed by an 'e', e.g.: ü = ue.
- France/Belgium: Omit accent de gues and accent de graphs, e.g.: ó = o.
- Sweden: Please change the last three letters of the Swedish alphabet and replace e.g.: ä = a.
- Poland: Please change any letter that cannot be found in the English alphabet, knowing that for some letters that one can only use a single letter instead of the three different letters used in the Polish alphabet, e.g.: z = z.
- Spain/Portugal: Please change any letter that cannot be found in the English alphabet, e.g.: ñ = n.
- Greece: Please do not use Greek letters. Please try to write the Greek words with Latin letters.
- Denmark: Please change any letter that contains non-English letters, e.g.: "å" with "aa".
- Slovakia, Czech Republic, Hungary, Rumania, Latvia, Lithuania, Estonia, Bulgaria, Slovenia, Croatia: PLEASE use your common sense, based on the examples from the other countries above.

10.6 Country name abbreviations

For convenience we provide a short list of names and two-digit codes (see Table 10.5) for the probably most important countries associated with the European Transmission Grid.

Table 10.5: Country codes

Country name	Country code	Country name	Country code
Albania	AL	Kosovo	XK
Armenia	AM	Latvia	LV
Austria	AT	Liechtenstein	LI
Azerbaijan	AZ	Lithuania	LT
Belarus	BY	Luxembourg	LU
Belgium	BE	Malta	MT
Bosnia and Herzegovina	BA	Moldova	MD
Bulgaria	BG	Montenegro	ME
Croatia	HR	Netherlands	NL
Cyprus	CY	Norway	NO
Czech	CZ	Poland	PL
Denmark	DK	Portugal	PT
Estonia	EE	Romania	RO
Finland	FI	Serbia	RS
France	FR	Slovakia	SK
Georgia	GE	Slovenia	SI
Germany	DE	Spain	ES
Greece	GR	Sweden	SE
Hungary	HU	Switzerland	CH
Iceland	IS	Turkey	TR
Ireland and Northern Ireland	IE	Belarus	UA
Italy	IT	Great Britain	GB
Russia Federation	RU	Europe	EU
Ukraine	UA		

10.7 IGGINLGE SciGRID_gas comparison with PDF source

Here, for all of Europe, the generated merged SciGRID_gas IGGINLGE data set will be shown in comparison with the original PDF source. It needs to be pointed out that for the country of Germany and the North Sea, the topological pipeline information cames from the LKD and NO data sources respectively.

10.7.1 Spain and Portugal

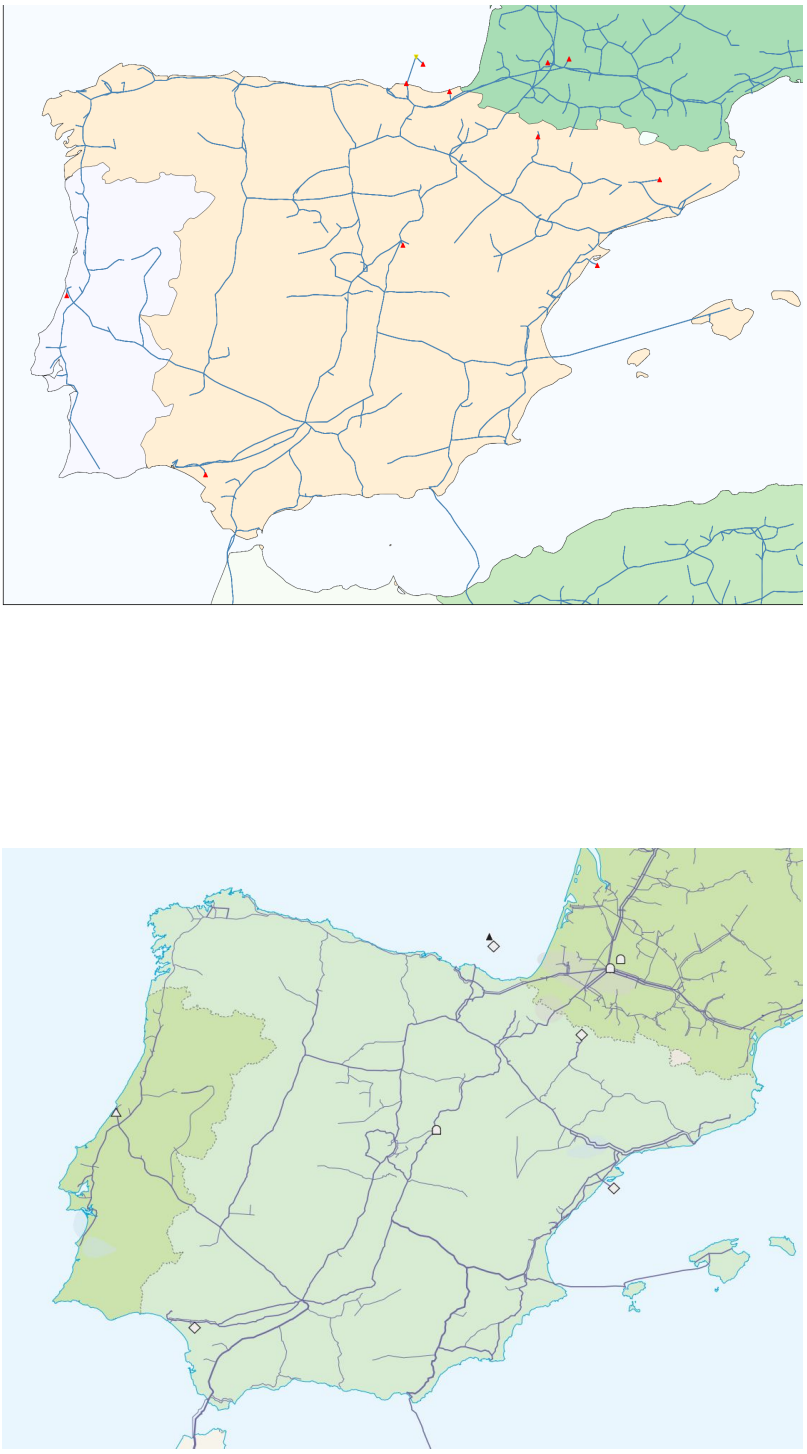


Figure 10.1: SciGRID_gas (top) and EntsoG (bottom) pipelines for Spain and Portugal.

10.7.2 France

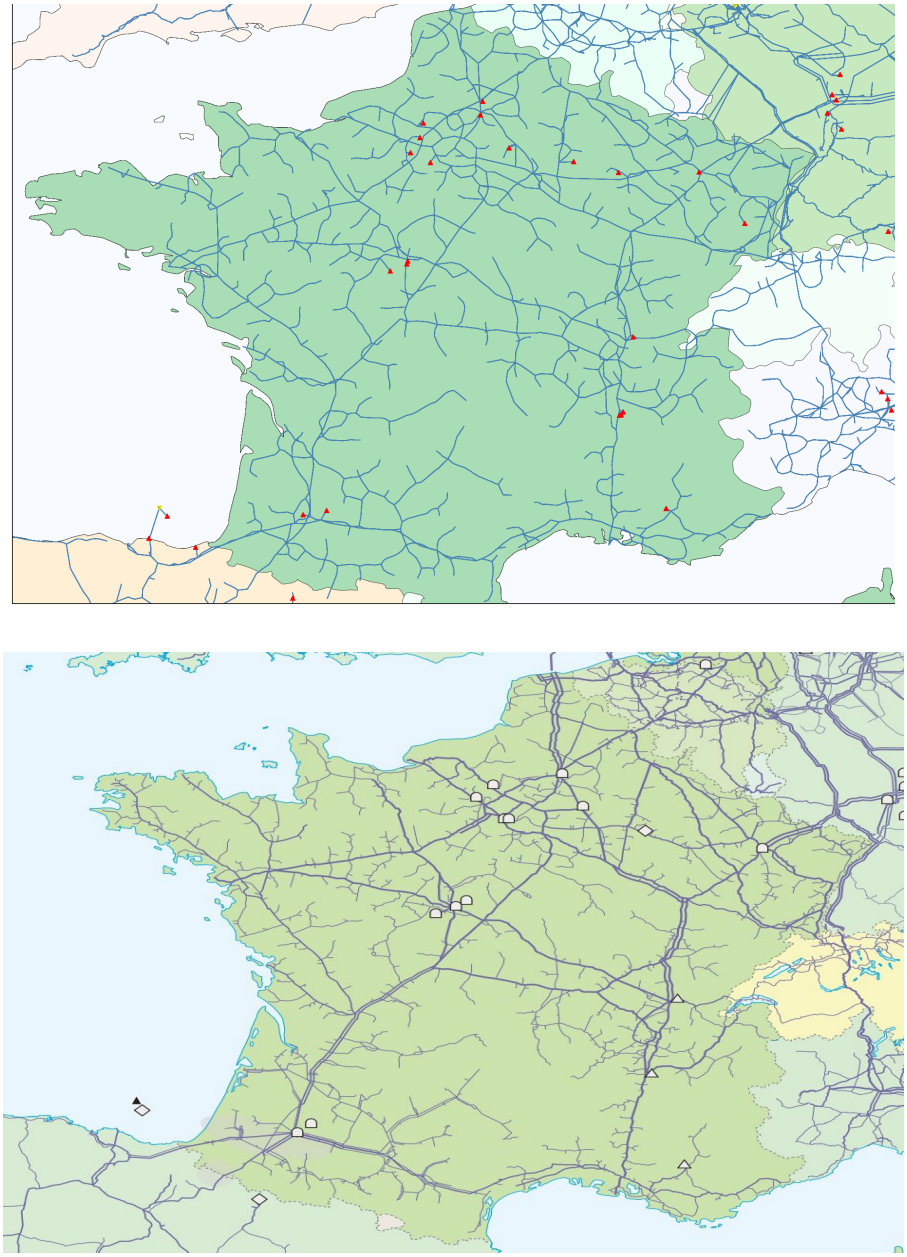


Figure 10.2: SciGRID_gas (top) and EntsoG (bottom) pipelines for France.

10.7.3 Germany

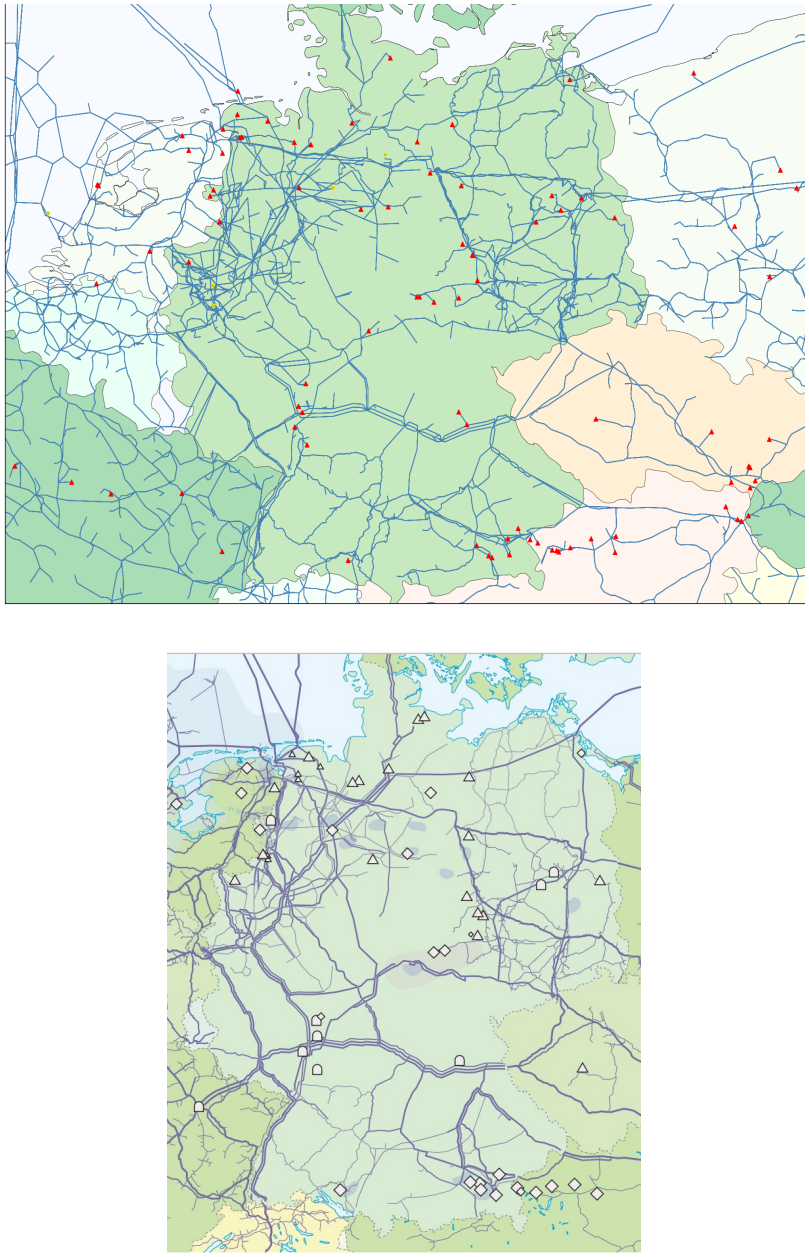


Figure 10.3: SciGRID_gas (top) and EntsoG (bottom) pipelines for Germany.

10.7.4 Belgium, Holland and Luxemburg

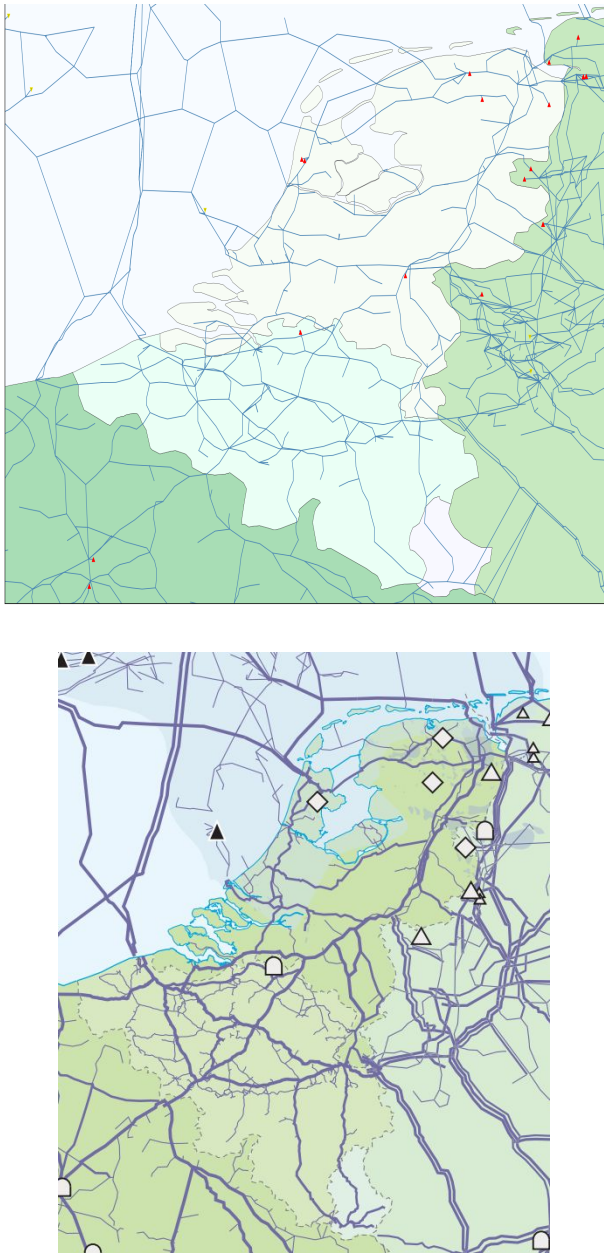


Figure 10.4: SciGRID_gas (top) and EntsoG (bottom) pipelines for Belgium, Holland and Luxemburg.

10.7.5 Austria, Czech Republic and Slowakia

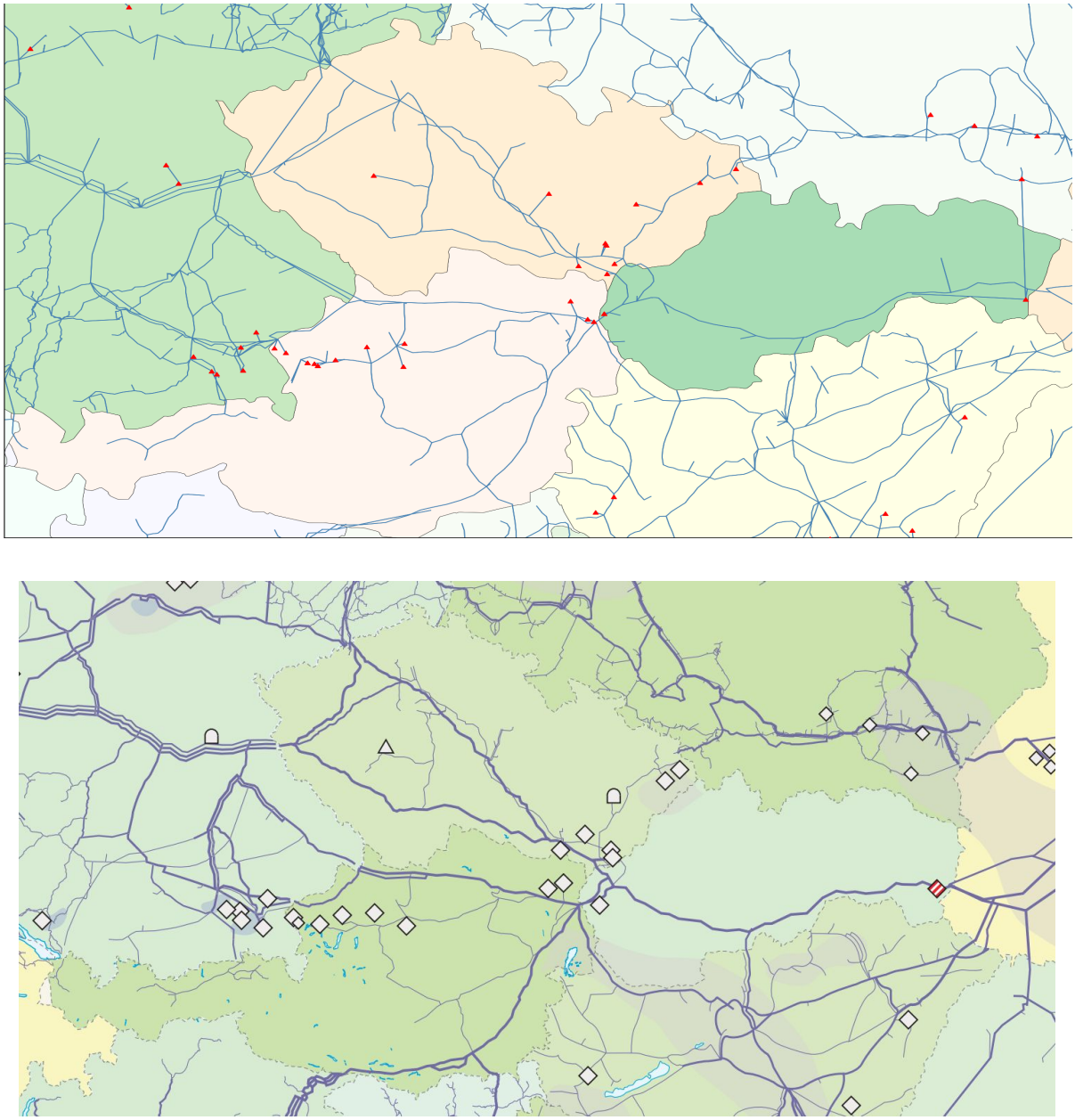


Figure 10.5: SciGRID_gas (top) and EntsoG (bottom) pipelines for Austria, Czech Republic and Slowakia.

10.7.6 Greece, Turkey and Bulgaria

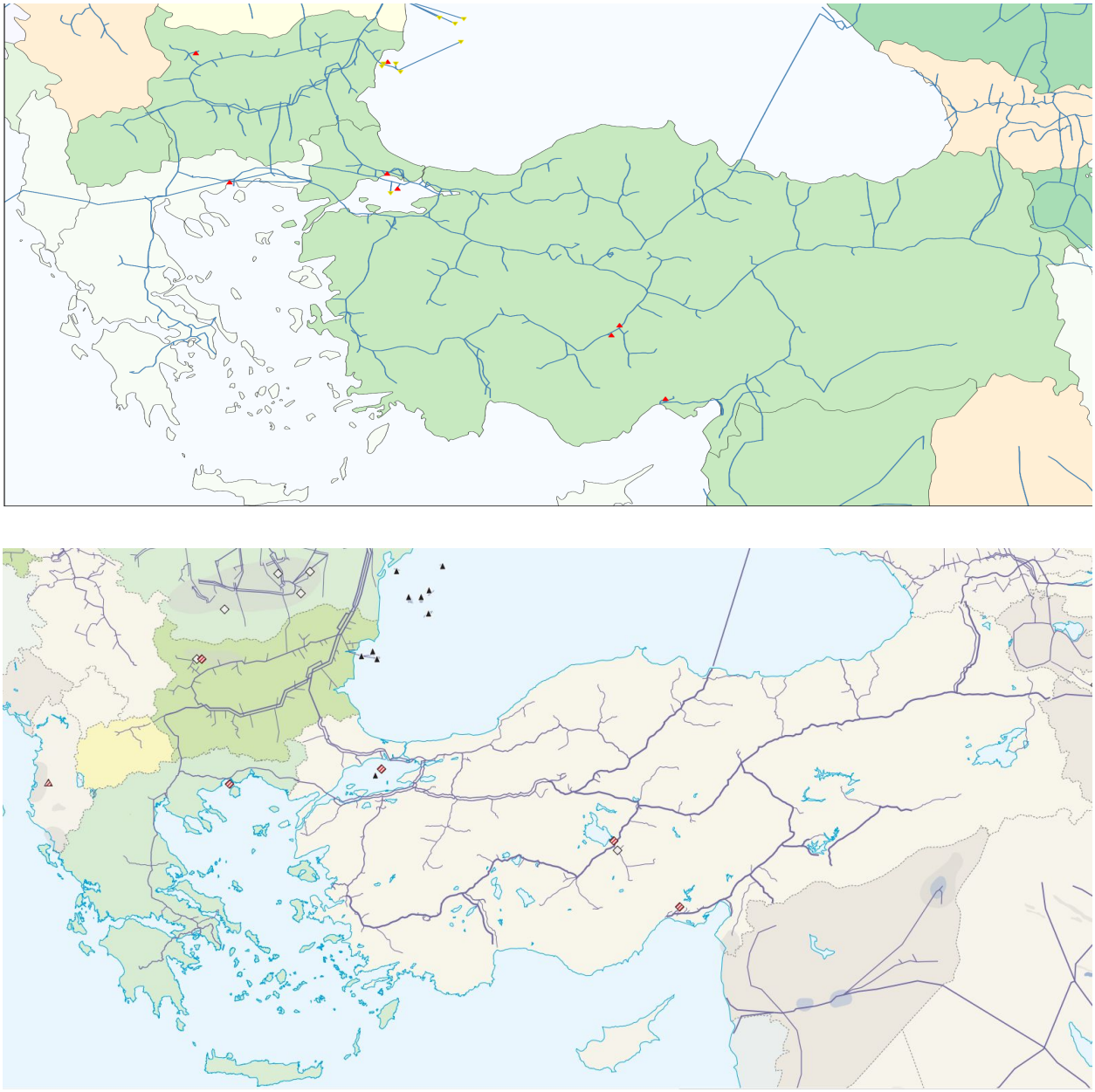


Figure 10.6: SciGRID_gas (top) and EntsoG (bottom) pipelines for Greece, Turkey and Bulgaria.

10.7.7 Italy

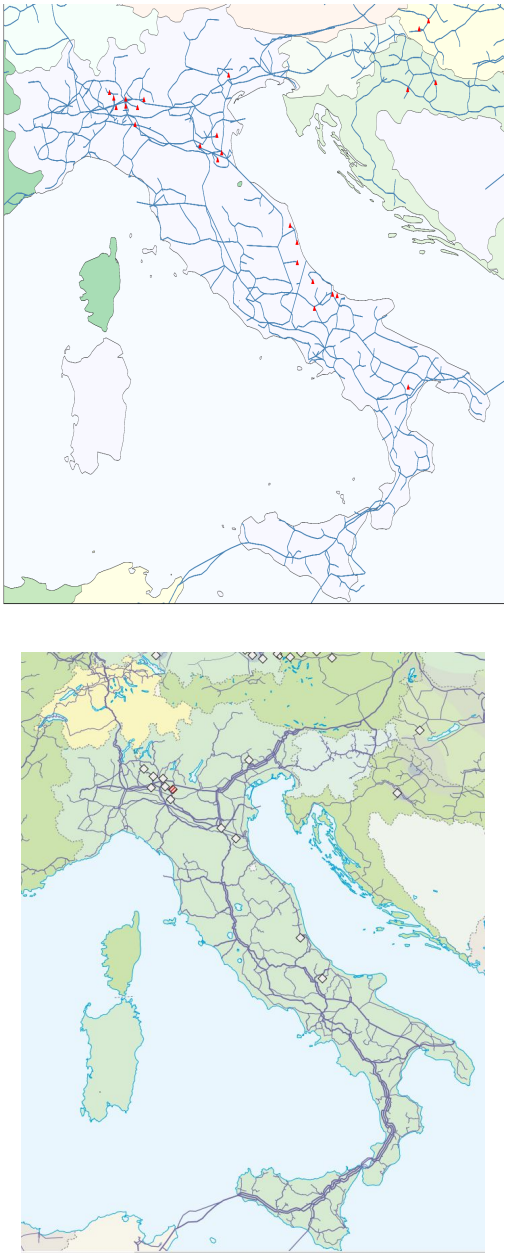


Figure 10.7: SciGRID_gas (top) and EntsoG (bottom) pipelines for Italy.

10.7.8 Ireland and UK

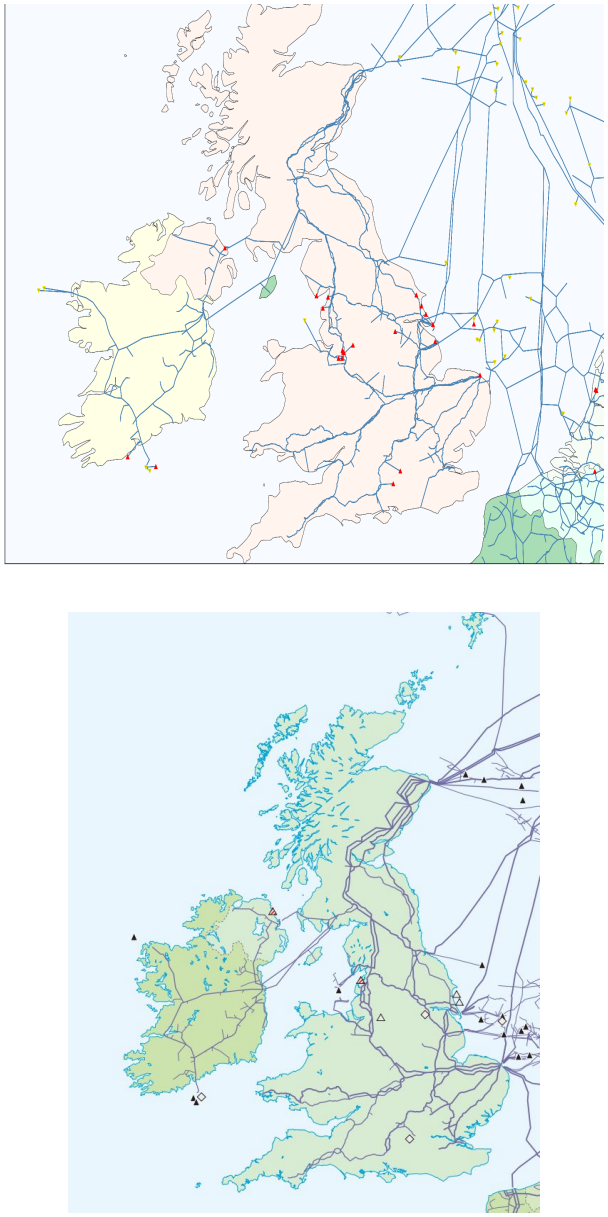


Figure 10.8: SciGRID_gas (top) and EntsoG (bottom) pipelines for Ireland and UK.

10.7.9 Poland

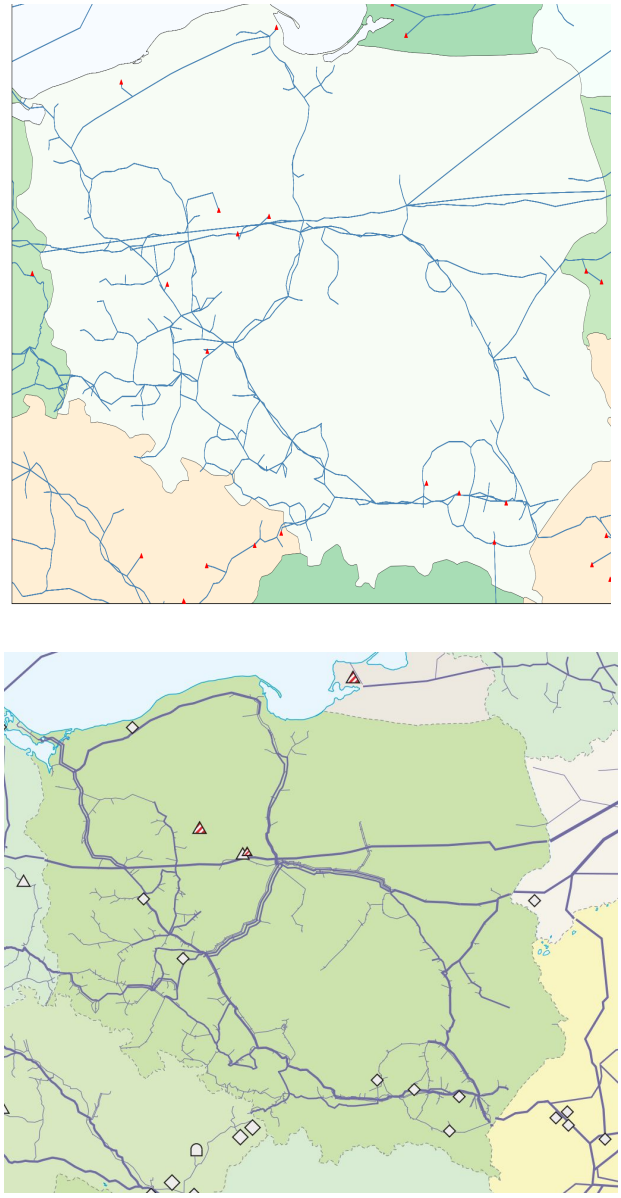


Figure 10.9: SciGRID_gas (top) and EntsoG (bottom) pipelines for Poland.

10.7.10 North Sea

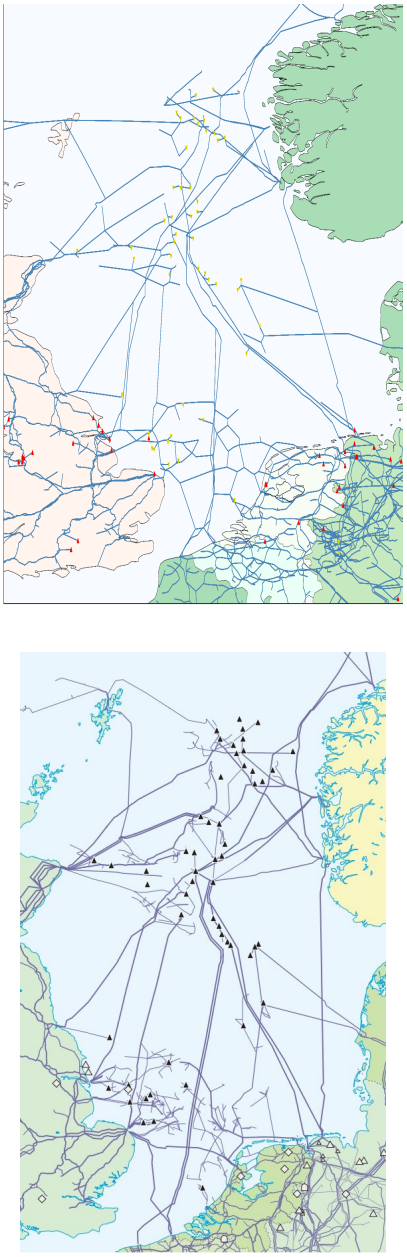


Figure 10.10: SciGRID_gas (top) and EntsoG (bottom) pipelines for the North Sea.

10.7.11 Baltic Sea

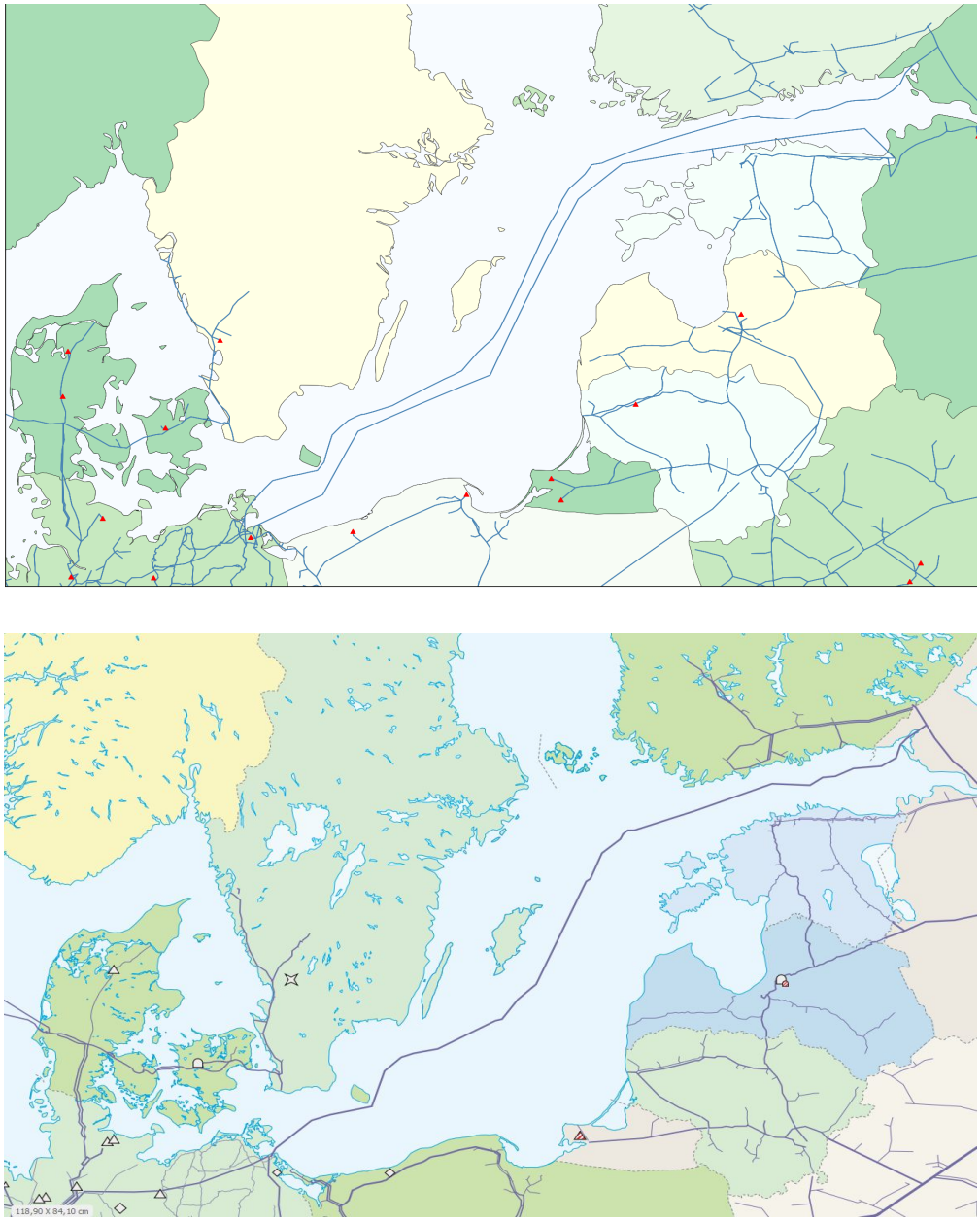


Figure 10.11: SciGRID_gas (top) and EntsoG (bottom) pipelines for the Baltic Sea.

10.7.12 Ukraine and Romania

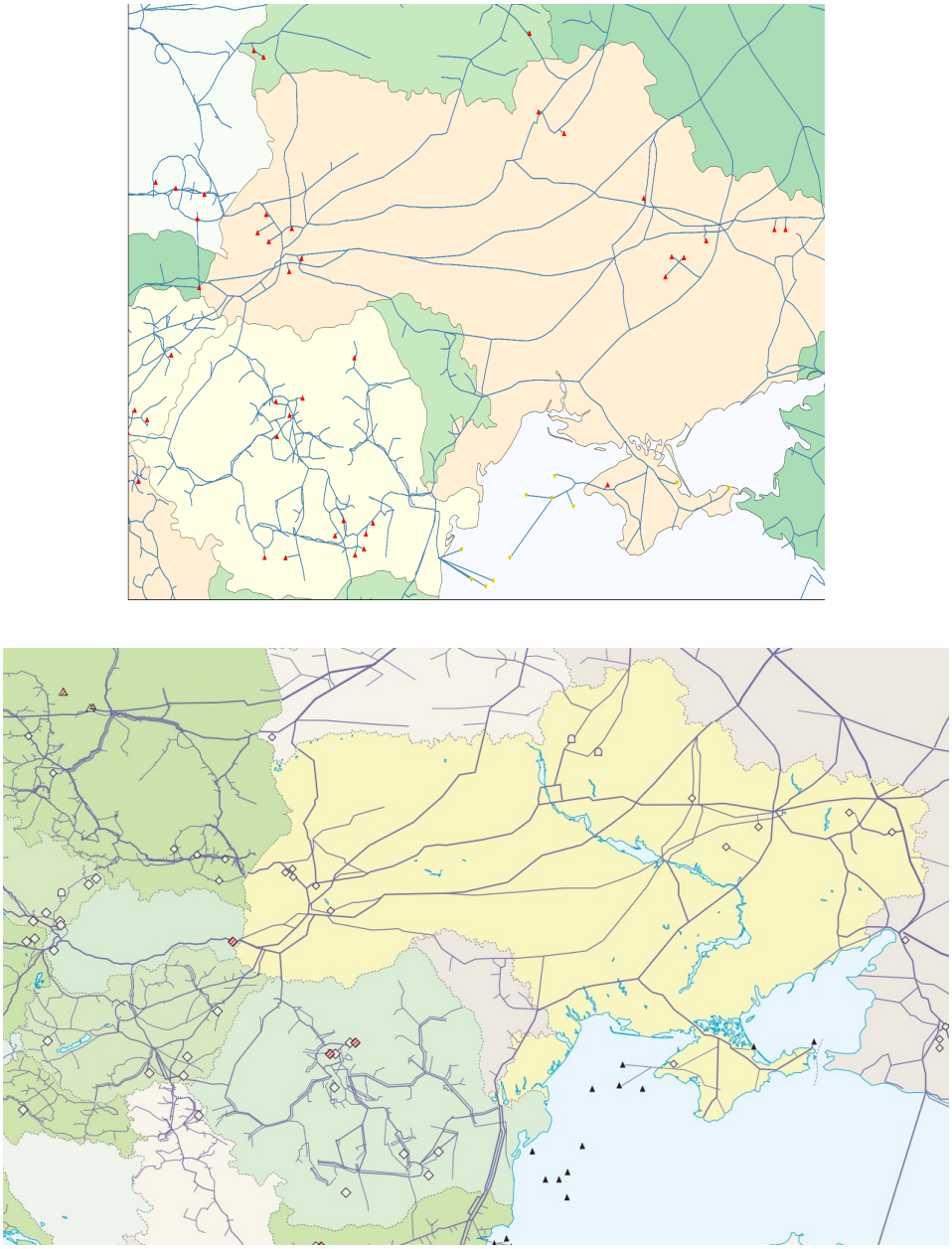


Figure 10.12: SciGRID_gas (top) and EntsoG (bottom) pipelines for Ukraine and Romania.

10.7.13 Belarus

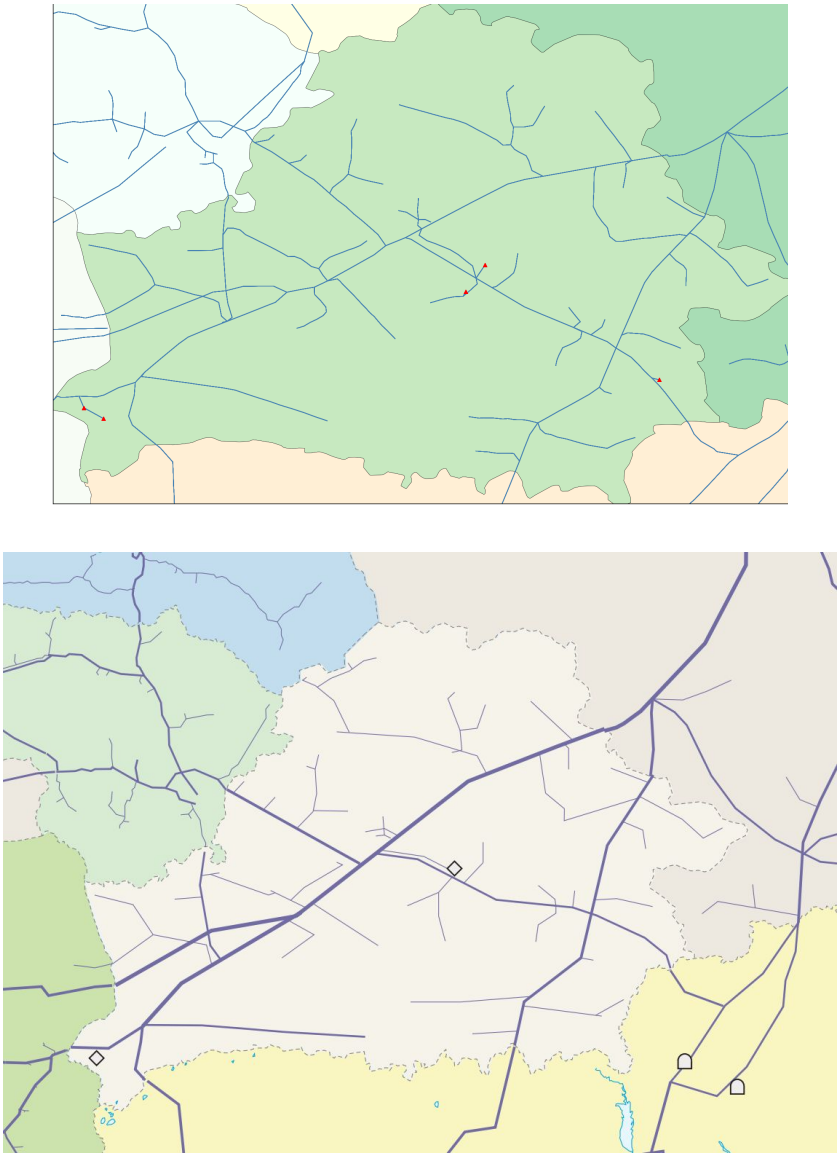


Figure 10.13: SciGRID_gas (top) and EntsoG (bottom) pipelines for Belarus.

10.7.14 Russia

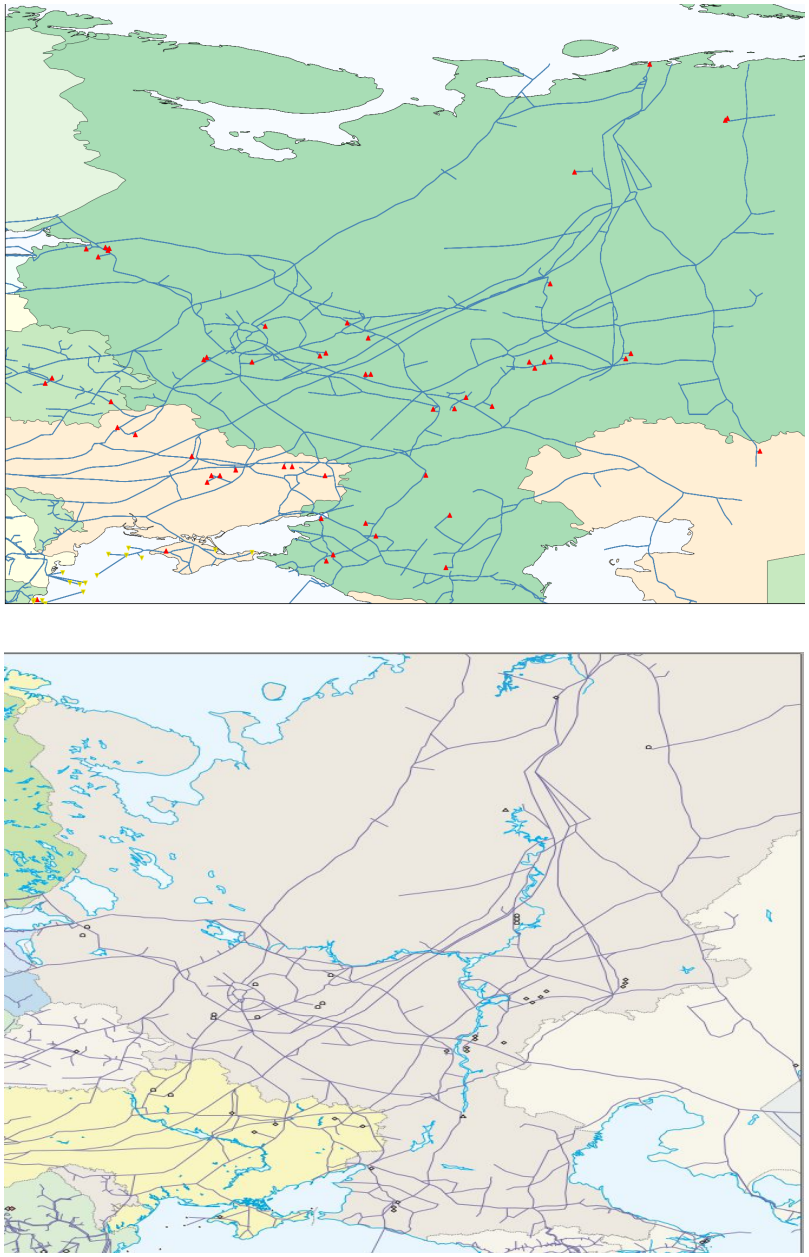


Figure 10.14: SciGRID_gas (top) and EntsoG (bottom) pipelines for European Russia.

10.7.15 East Africa

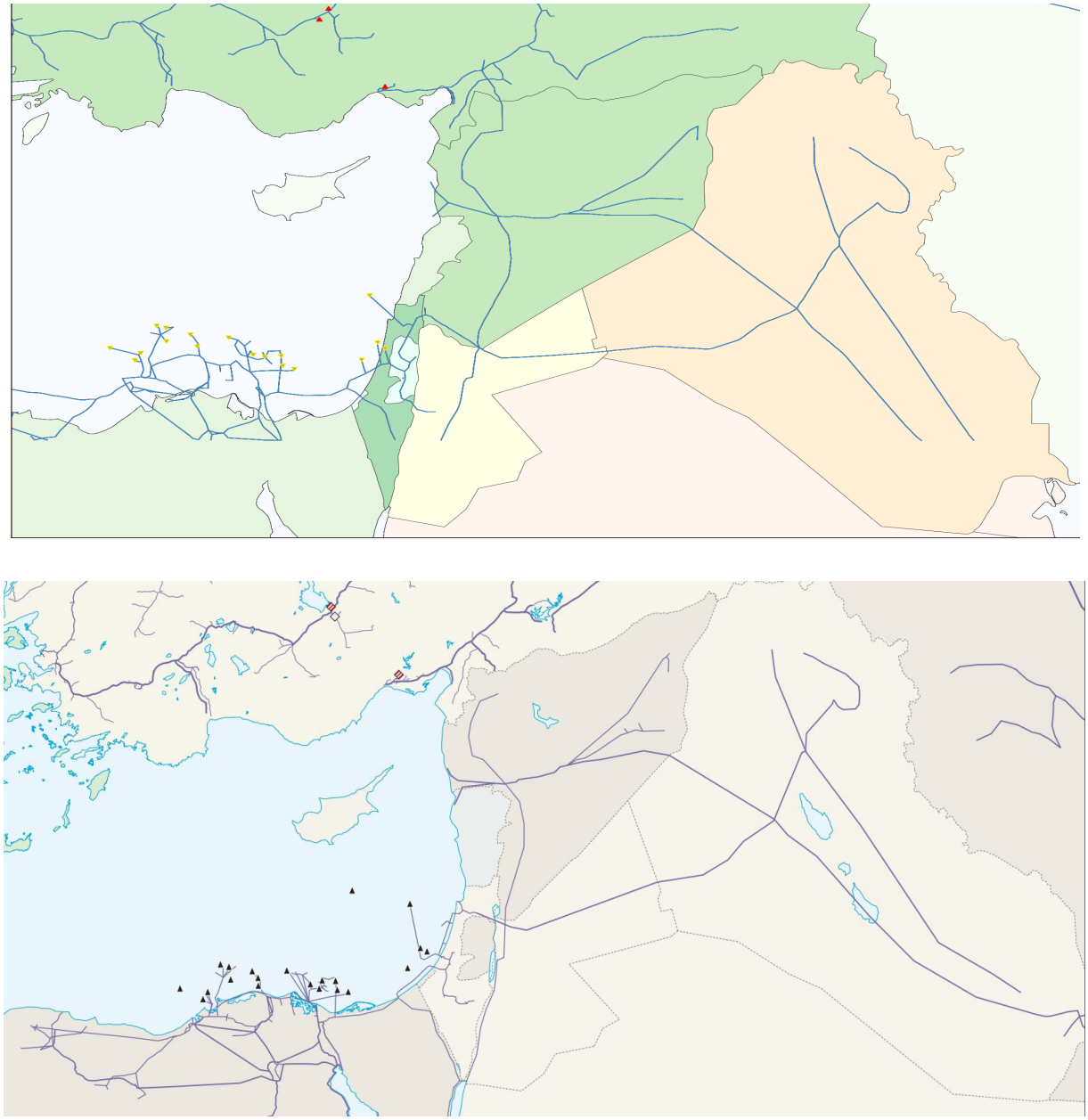


Figure 10.15: SciGRID_gas (top) and EntsoG (bottom) pipelines for Eastern Africa.

10.7.16 West Africa

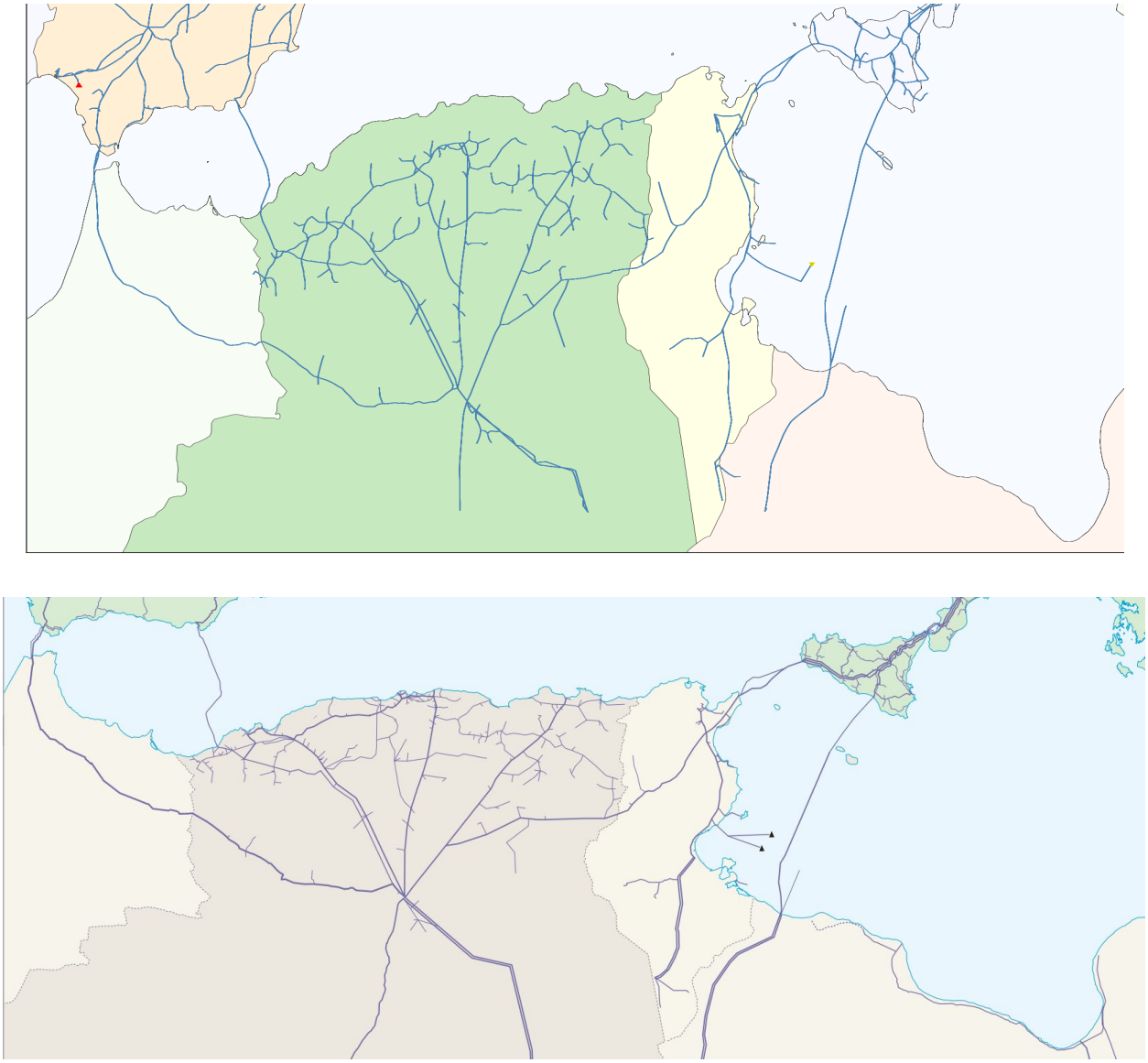


Figure 10.16: SciGRID_gas (top) and EntsoG (bottom) pipelines for Western Africa.

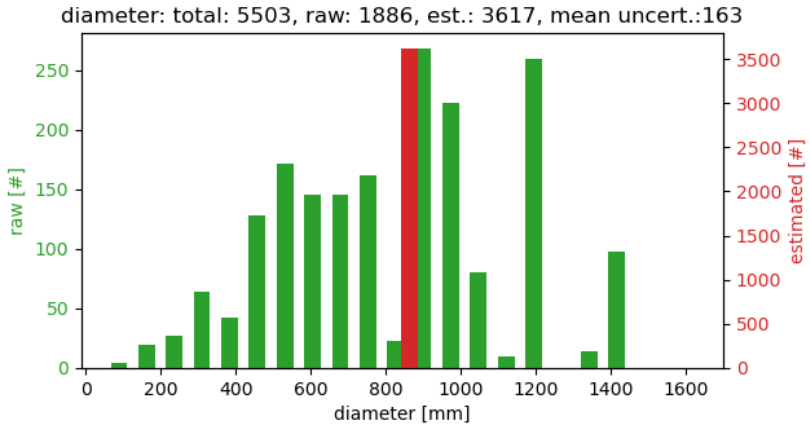
10.8 Heuristic histogram plots of the IGGIELGNC-3 data set

Below, for each filled attribute two histogram plots will be presented. The first plot for each attribute will be the histogram with normal Y-axis scaling, whereas the second plot will depict the histogram on a log Y-axis scale. Each of those plots will show in green bars the histogram of the raw input data (left Y-axis), and with red bars the histogram of the estimated values (right Y-axis). The title contains the name of the attribute, the total number of elements of this attribute, the number of raw input values, the total sum of generated attribute values and the overall mean uncertainty of the attribute values. In addition, below each graph with the linear scale, a list of methods used is given. Each method name is followed by the name of the independent variable or variables, supplied in brackets. This is followed by the number of attribute values that were generated with the methods. The values in the last bracket in each line give the mean uncertainty for those attribute values generated, excluding the raw input data. Further information is given in the last line, where the minimum and maximum values of the raw input data ("min(raw)" and "max(raw)"), and the minimum and maximum of the estimated data ("min(raw)" and "max(raw)") are presented.

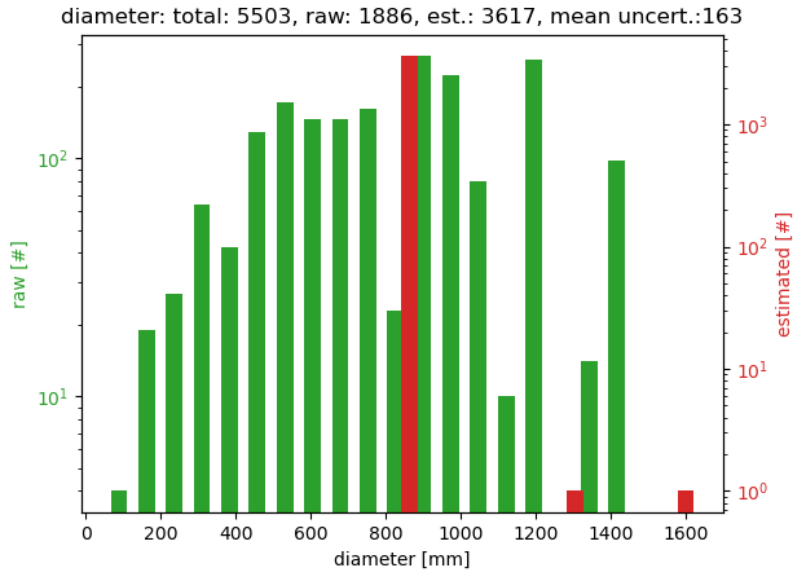
10.8.1 *PipeSegments*

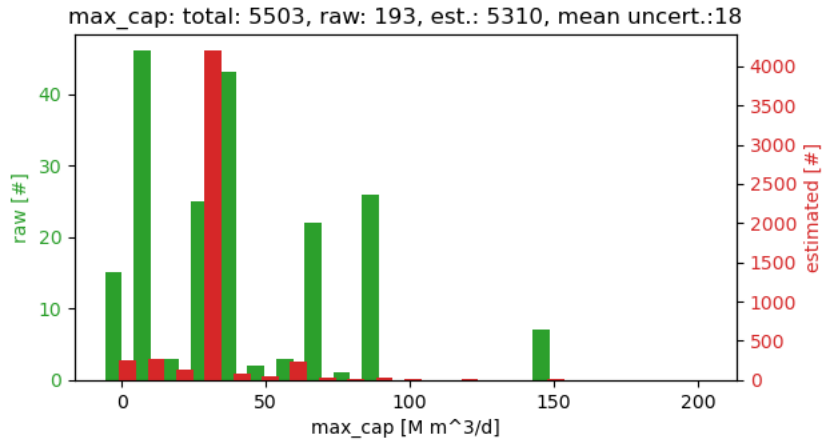
Below are the heuristic histogram plots of the component *PipeSegments* for the attributes:

- *diameter_mm*
- *max_cap_M_m3_per_d*
- *max_pressure_bar*.

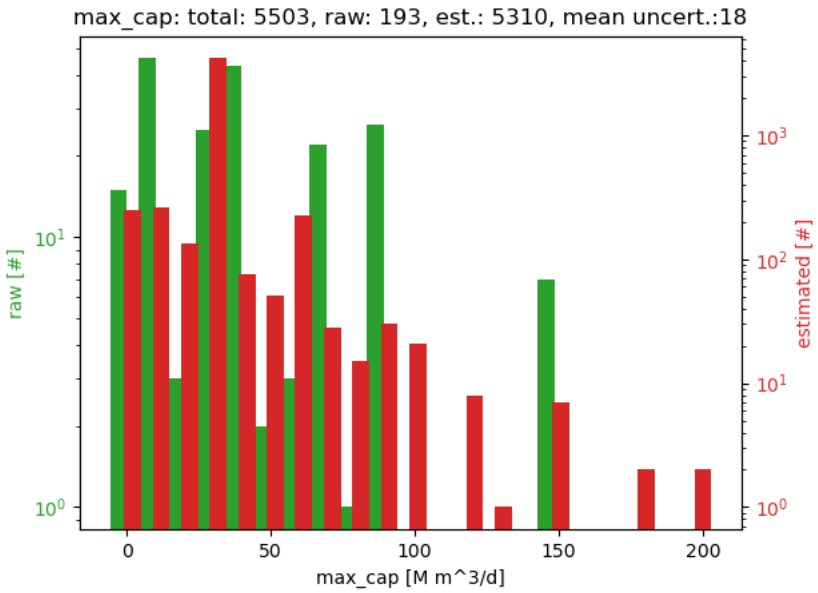


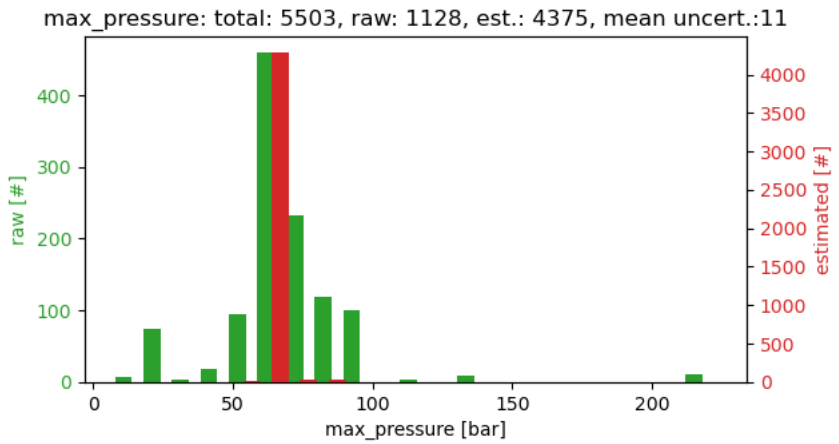
Median(diameter_mm): 3614 (mean uncert.: 247)
M_Heuristics.adjust_AttribValues(): 1 (mean uncert.: 500)
log_M_Heuristics.PipeSegment_diameter(): 2 (mean uncert.: 132)
min(raw):125.0, max(raw):1422.4, min(est):863.6, max(est):1600.0



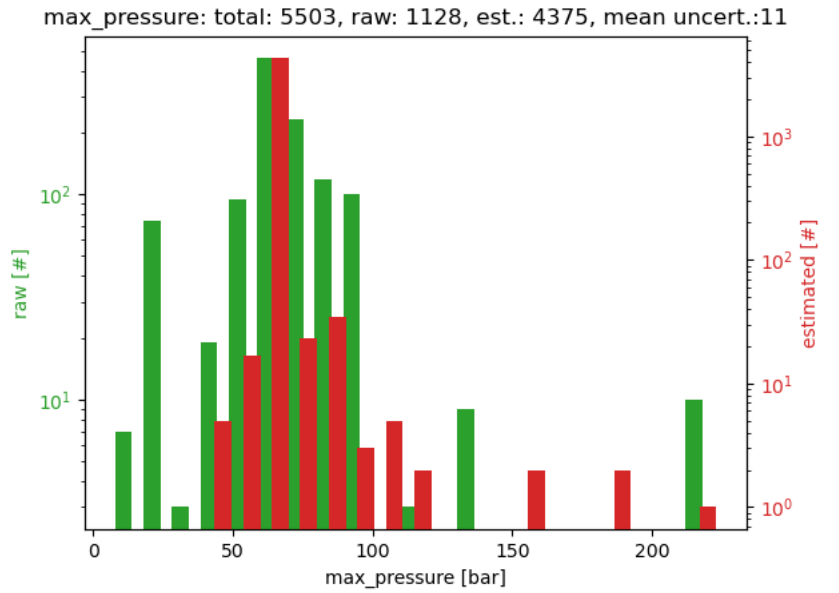


Median(max_cap_M_m3_per_d): 4050 (mean uncert.: 24)
M_Heuristics.app_Phys_Heur_Move_Capacity(): 238 (mean uncert.: 0)
log_M_Heuristics.PipeSegment_capacity(): 404 (mean uncert.: 1)
M_Heuristics.adjust_AttribValues(): 24 (mean uncert.: 50)
log_raw: 592 (mean uncert.: 0)
Squared(Compressors_max_power_MW, Storages_max_cushionGas_M_m3): 2 (mean uncert.: 1)
min(raw):6.3, max(raw):150.68, min(est):2.0, max(est):200.0





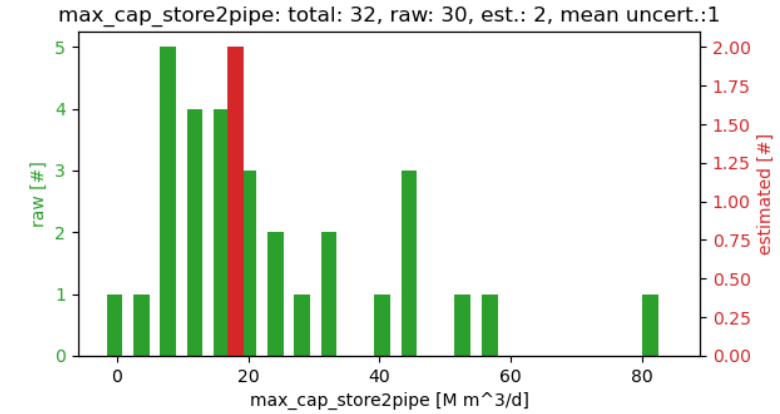
Median(max_pressure_bar): 4264 (mean uncert.: 14)
log_M_Heuristics.PipeSegment_pressure(): 106 (mean uncert.: 1)
M_Heuristics.adjust_AttribValues(): 1 (mean uncert.: 100)
Squared(Compressors_max_power_MW, Storages_max_power_MW): 3 (mean uncert.: 1)
Lasso(Compressors_max_power_MW, Storages_max_cushionGas_M_m3): 1 (mean uncert.: 1)
min(raw):16.0, max(raw):220.0, min(est):43.0, max(est):220.0

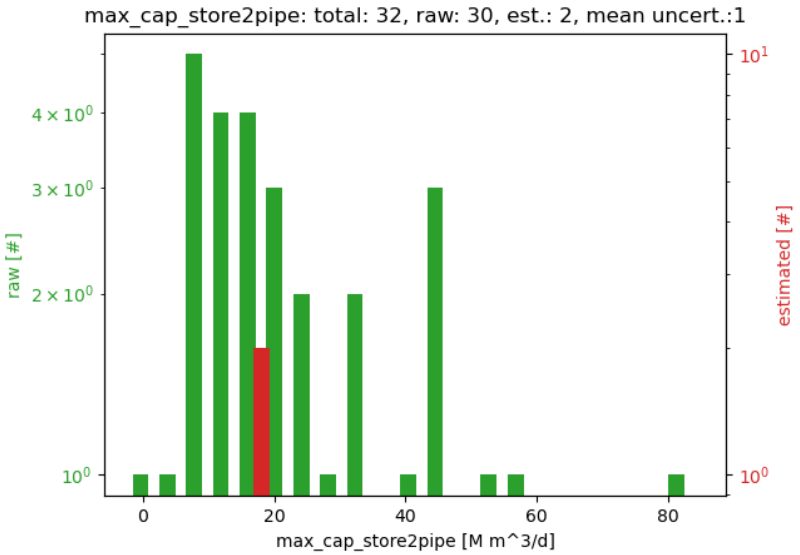


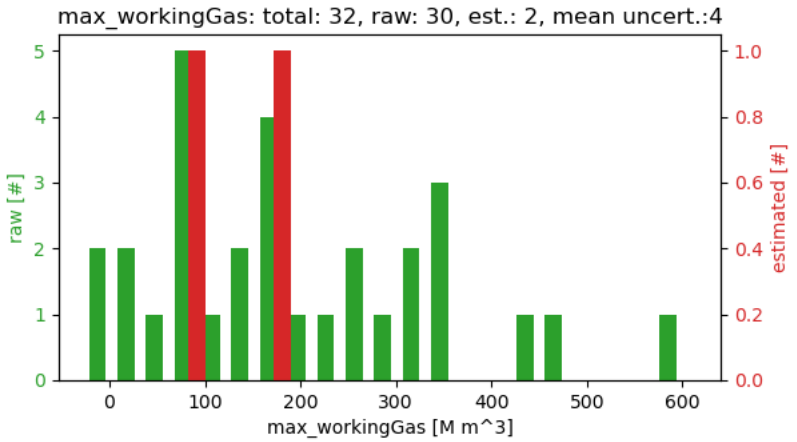
10.8.2 *LNGs*

Below are the heuristic histogram plots of the component *LNGs* for the attributes:

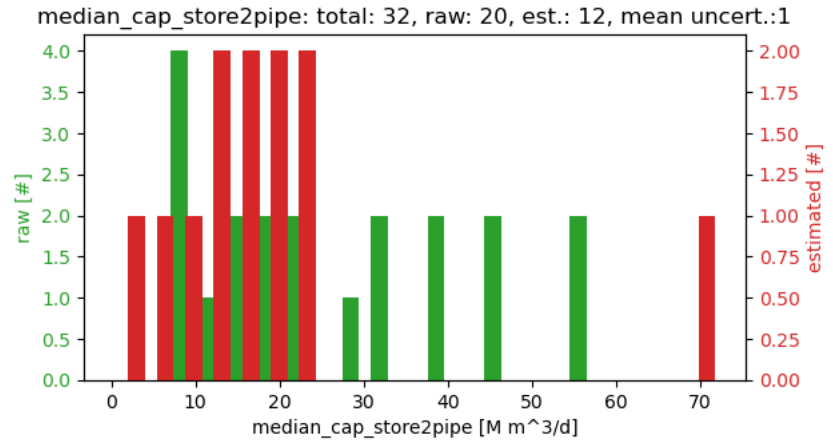
- *max_cap_store2pipe_M_m3_per_d*
- *max_workingGas_M_m3*
- *median_cap_store2pipe_M_m3_per_d.*



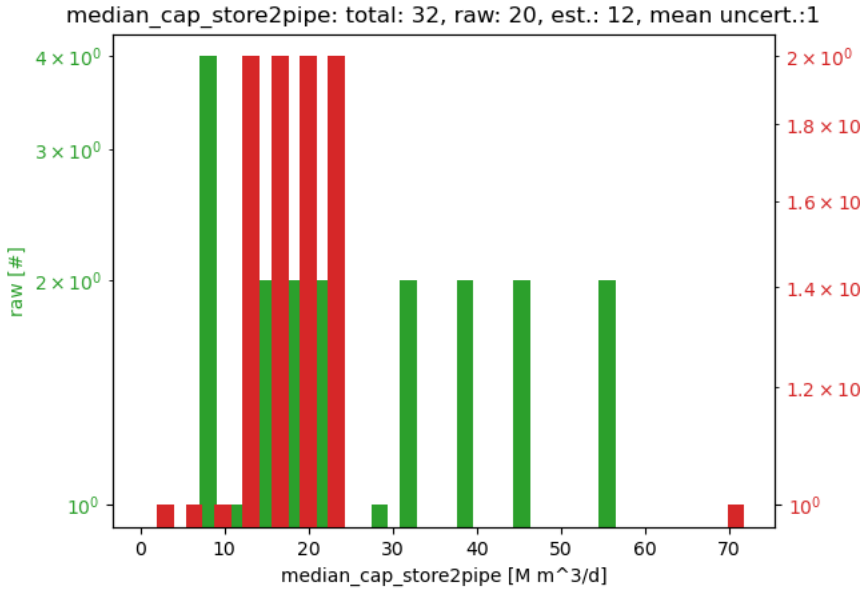




Median(max_workingGas_M_m3): 1 (mean uncert.: 121)
Squared(max_cap_store2pipe_M_m3_per_d, Pipe_max_cap_M_m3_per_d): 1 (mean uncert.: 8)
min(raw):2.32, max(raw):599.42, min(est):104.47, max(est):175.2



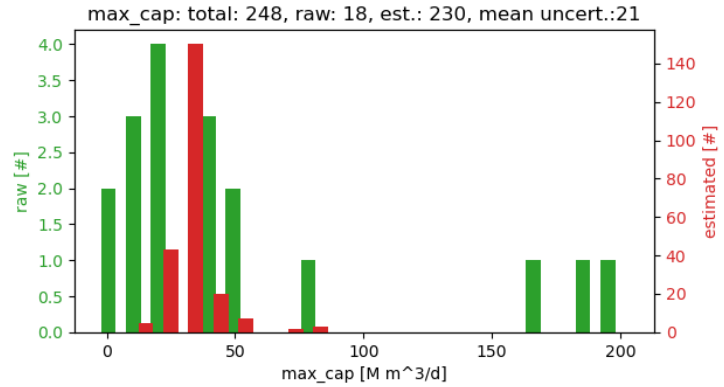
Lasso(max_cap_store2pipe_M_m3_per_d, max_workingGas_M_m3): 8 (mean uncert.: 2)
Median(median_cap_store2pipe_M_m3_per_d): 2 (mean uncert.: 13)
Lasso(max_cap_store2pipe_M_m3_per_d): 2 (mean uncert.: 2)
min(raw):8.44, max(raw):57.18, min(est):3.01, max(est):70.8

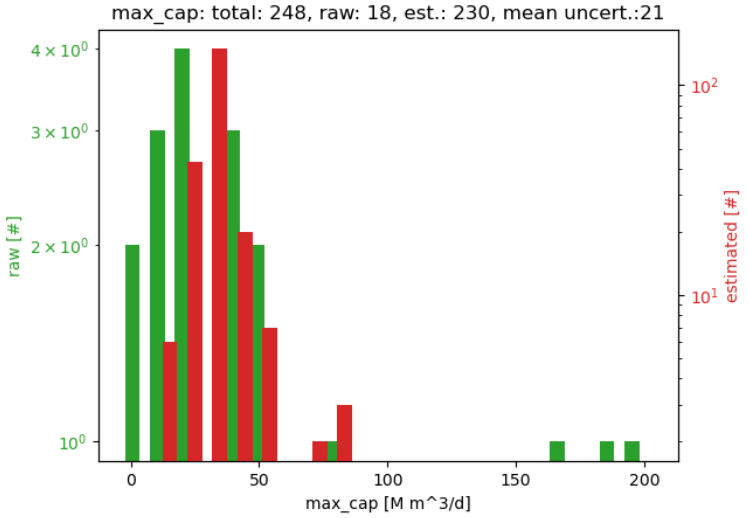


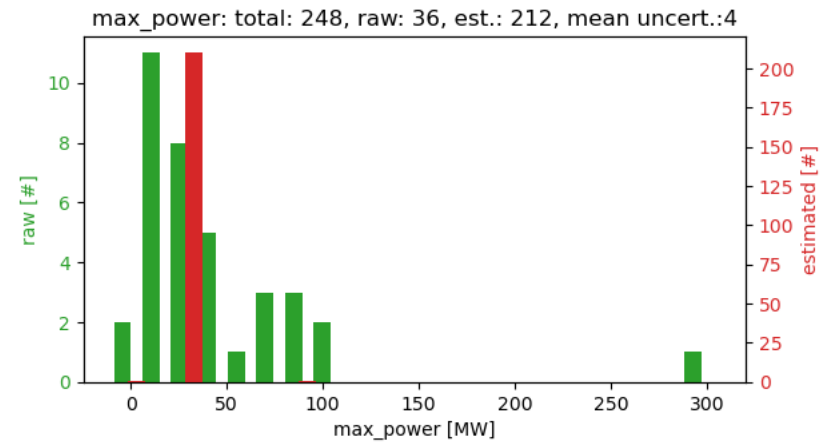
10.8.3 *Compressors*

Below are the heuristic histogram plots of the component *Compressors* for the attributes:

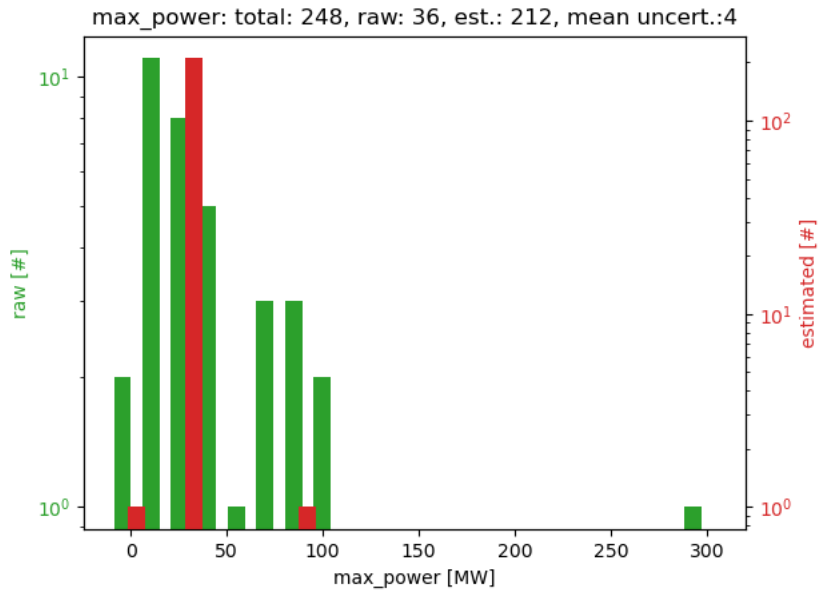
- *max_cap_M_m3_per_d*
- *max_power_MW*
- *max_pressure_bar*
- *num_turb*
- *turbine_power_1_MW*
- *turbine_power_2_MW*
- *turbine_power_3_MW*
- *turbine_power_4_MW*.

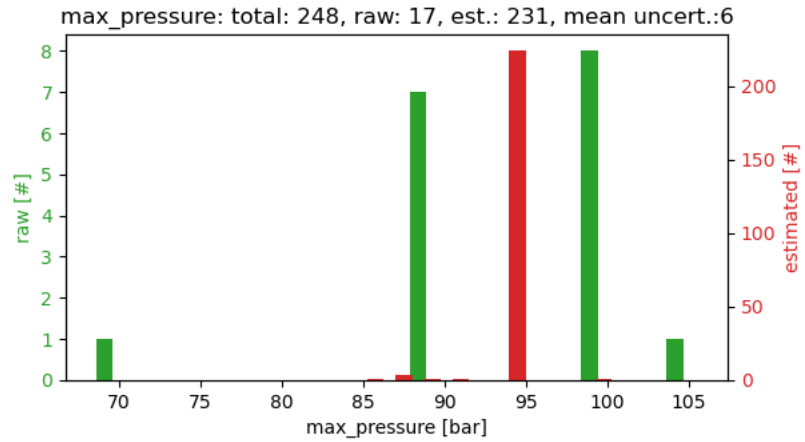




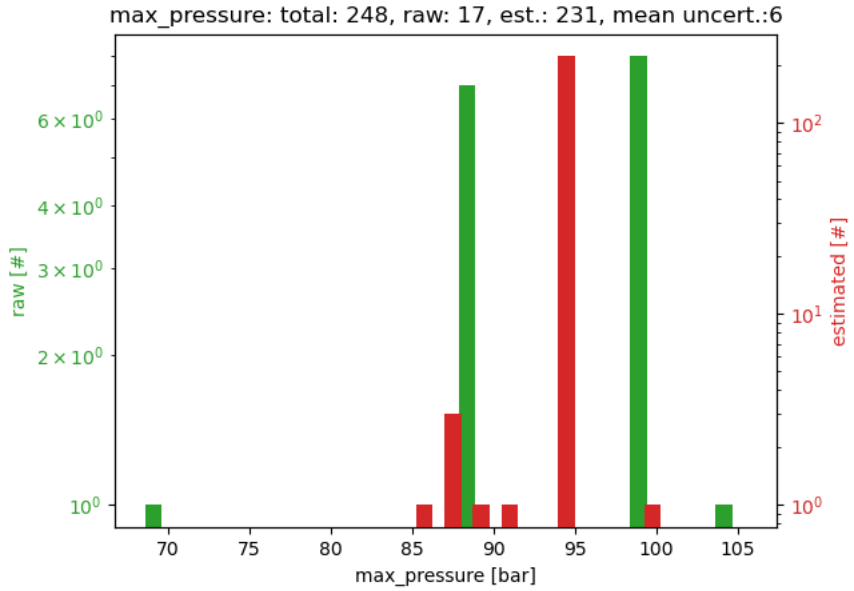


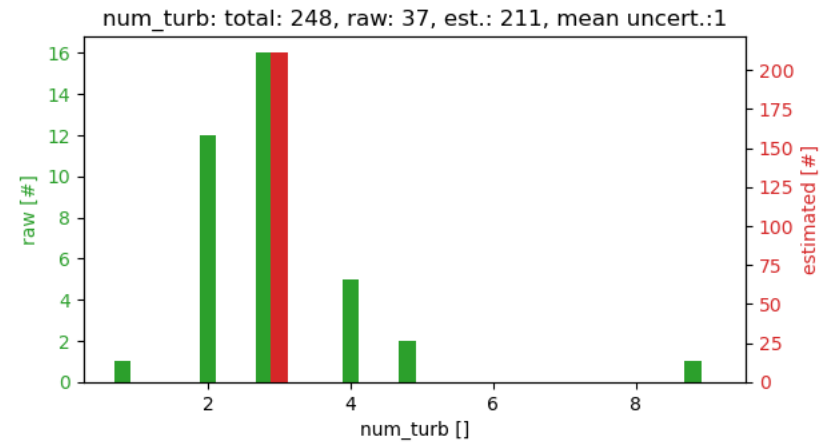
Lasso(num_turb, turbine_power_4_MW): 208 (mean uncert.: 5)
M_Internet.set_max_power_MW: 4 (mean uncert.: 0)
min(raw):5.5, max(raw):300.0, min(est):3.0, max(est):85.6



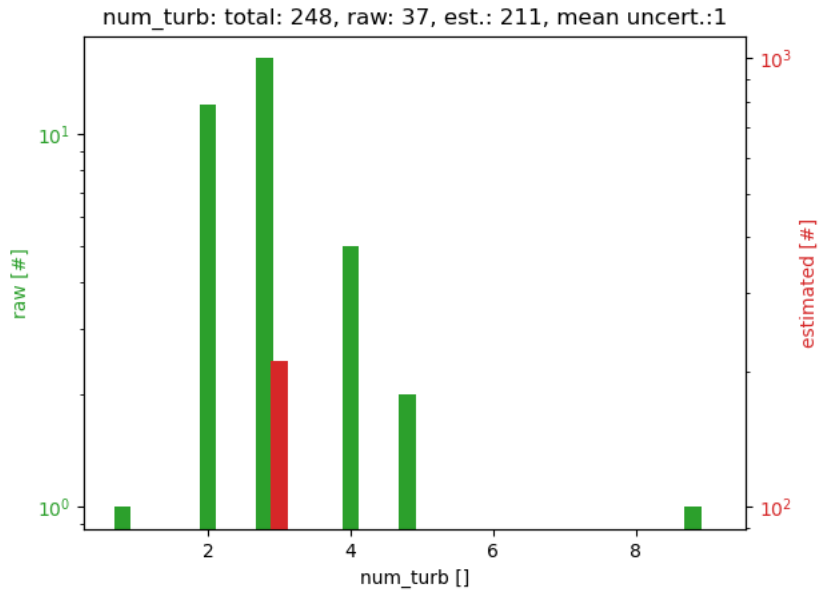


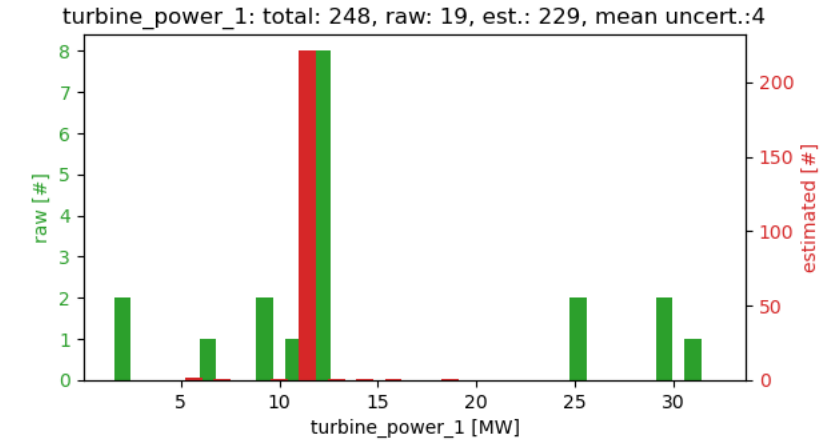
Median(max_pressure_bar): 224 (mean uncert.: 6)
Lasso(turbine_power_2_MW, Pipe_max_cap_M_m3_per_d): 3 (mean uncert.: 4)
Lasso(turbine_power_3_MW, Pipe_max_cap_M_m3_per_d): 4 (mean uncert.: 3)
min(raw):70.0, max(raw):105.0, min(est):86.19, max(est):100.59



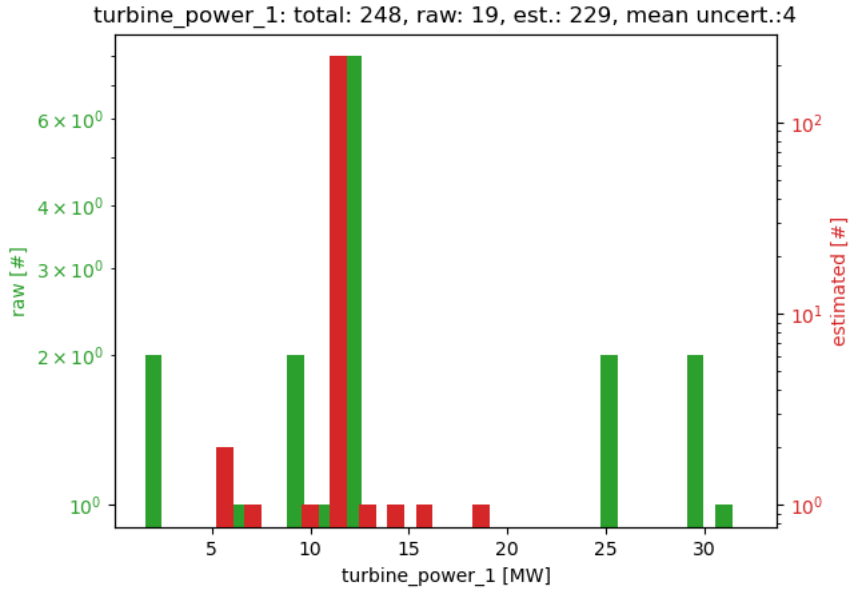


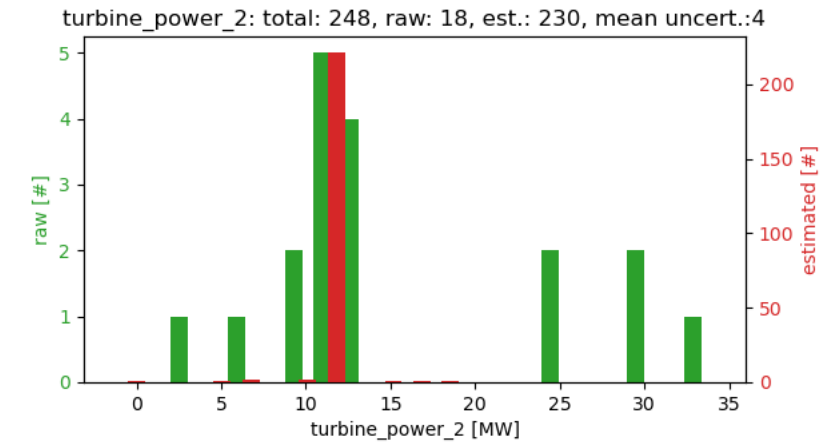
Median(num_turb): 211 (mean uncert.: 1)
min(raw):1, max(raw):9, min(est):3, max(est):3



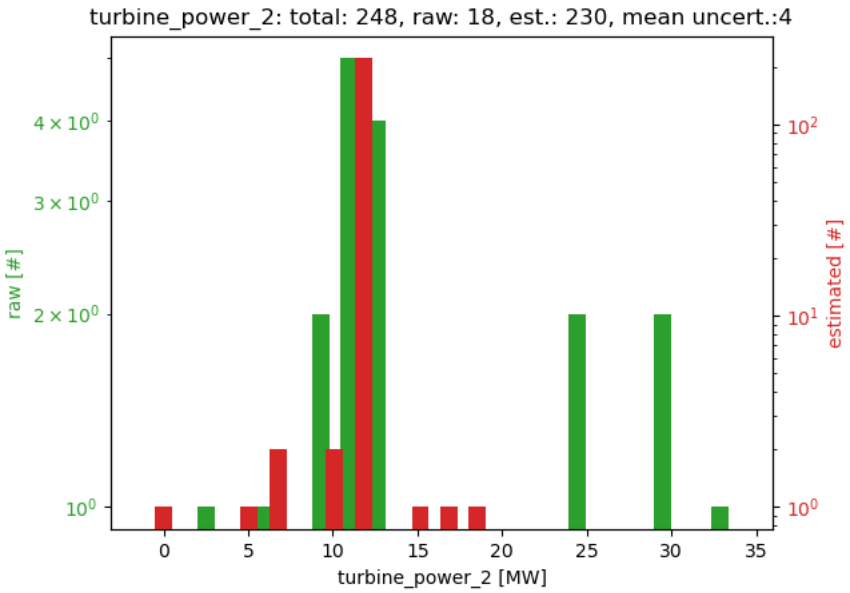


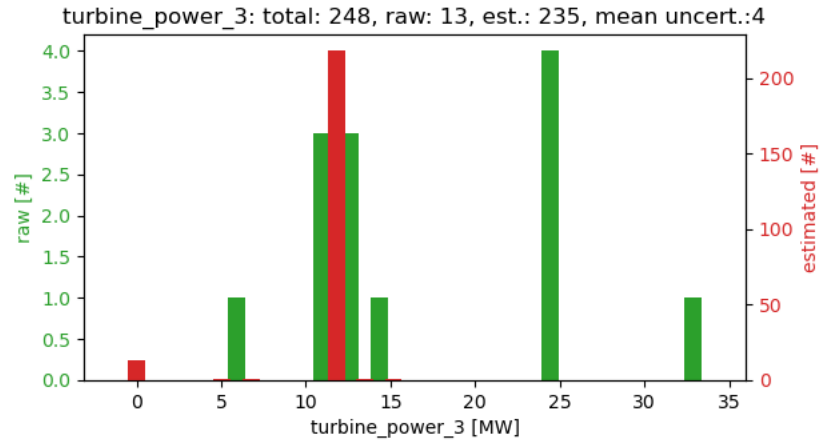
Median(turbine_power_1_MW): 212 (mean uncert.: 4)
M_Internet.set_turbine_power: 17 (mean uncert.: 6)
min(raw):2.75, max(raw):31.7, min(est):5.0, max(est):19.3



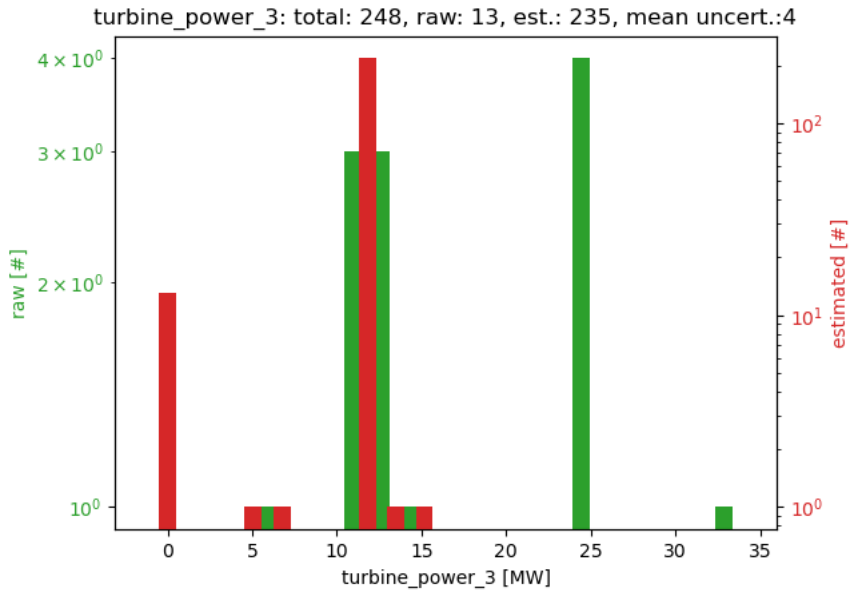


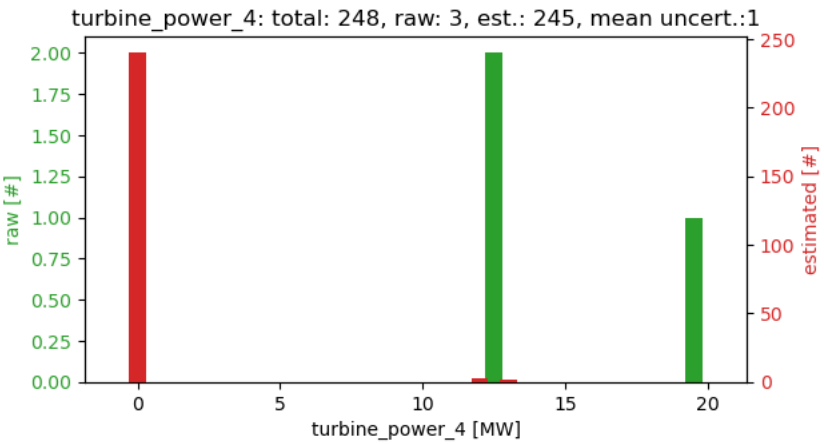
Median(turbine_power_2_MW): 212 (mean uncert.: 4)
M_Internet.set_turbine_power: 17 (mean uncert.: 6)
make_Attrib(const): 1 (mean uncert.: 0)
min(raw):2.75, max(raw):33.7, min(est):0, max(est):19.3



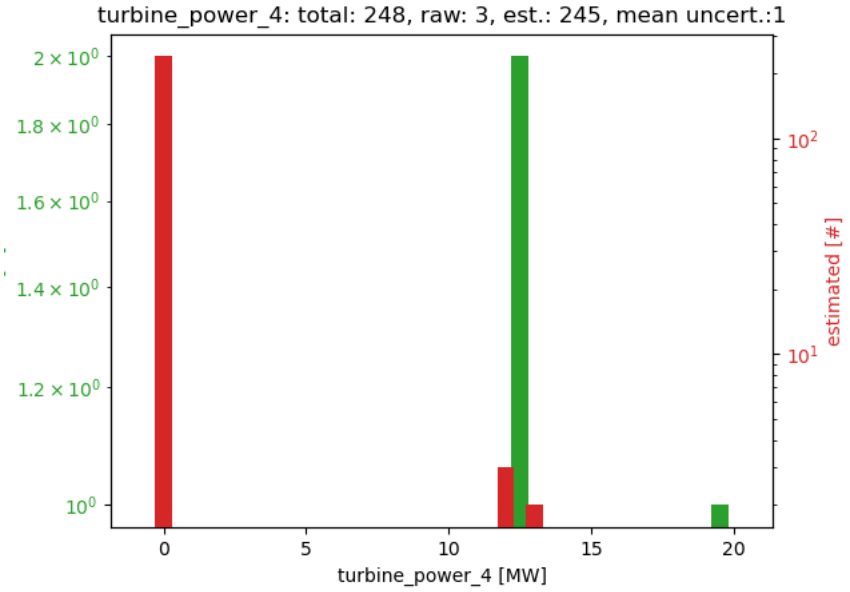


Median(turbine_power_3_MW): 212 (mean uncert.: 4)
make_Attrib(const): 13 (mean uncert.: 0)
M_Internet.set_turbine_power: 9 (mean uncert.: 5)
copied from turbine_power_2_MW: 1 (mean uncert.: 0)
min(raw):7.0, max(raw):33.7, min(est):0, max(est):15.0





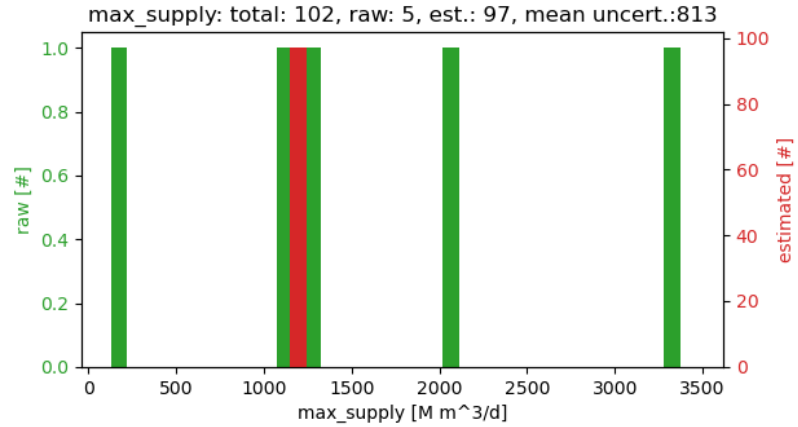
make_Attrib(const): 240 (mean uncert.: 0)
M_Internet.set_turbine_power: 4 (mean uncert.: 6)
copied from turbine_power_3_MW: 1 (mean uncert.: 0)
min(raw):12.5, max(raw):20.0, min(est):0, max(est):13.0



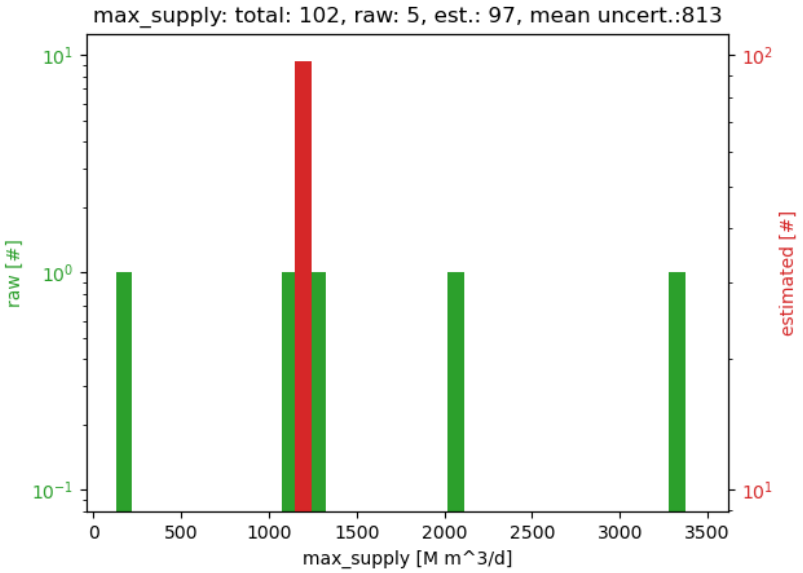
10.8.4 *Productions*

Below are the heuristic histogram plots of the component *Productions* for the attributes:

- *max_supply_M_m3_per_d*.



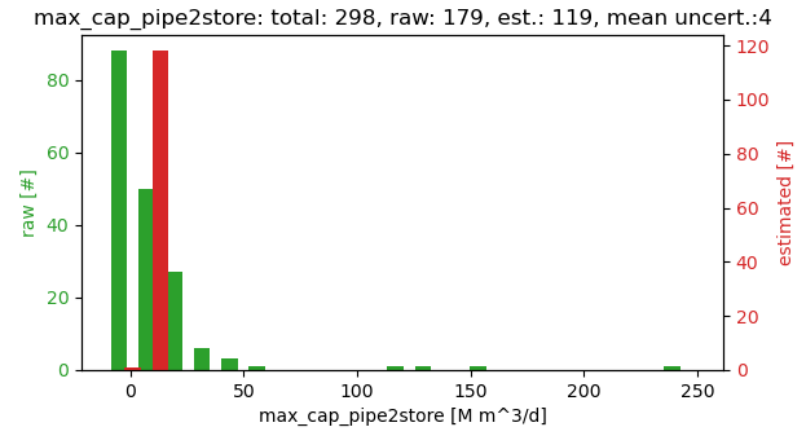
Median(max_supply_M_m3_per_d): 97 (mean uncert.: 855)
min(raw):253.66, max(raw):3403.59, min(est):1230.41, max(est):1230.41



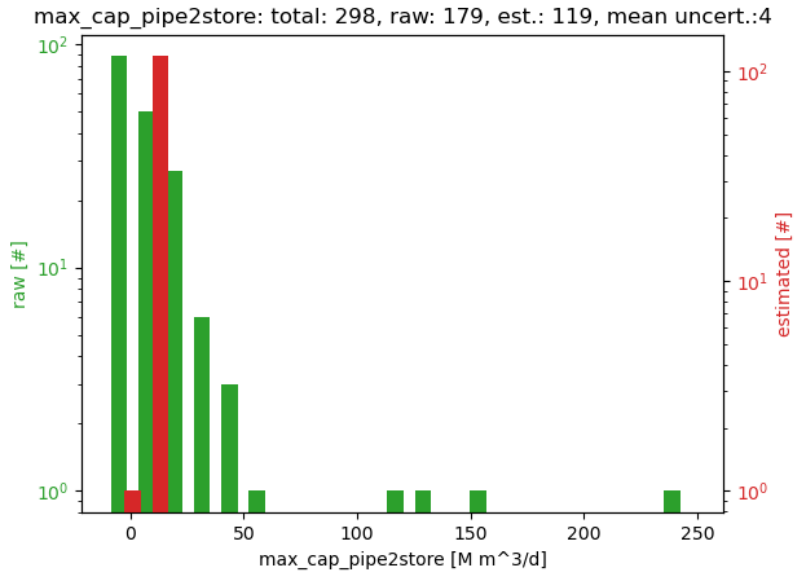
10.8.5 *Storages*

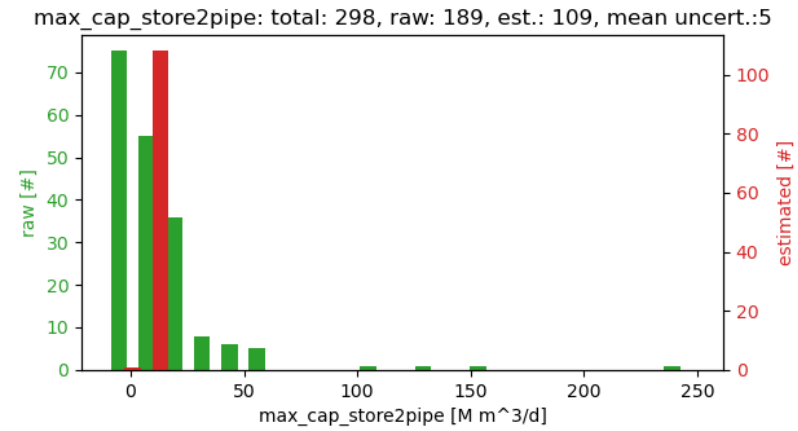
Below are the heuristic histogram plots of the component *Storages* for the attributes:

- *max_cap_pipe2store_M_m3_per_d*
- *max_cap_store2pipe_M_m3_per_d*
- *max_cushionGas_M_m3*
- *max_power_MW*
- *max_storage_pressure_bar*
- *max_workingGas_M_m3*
- *min_storage_pressure_bar*
- *num_storage_wells*.

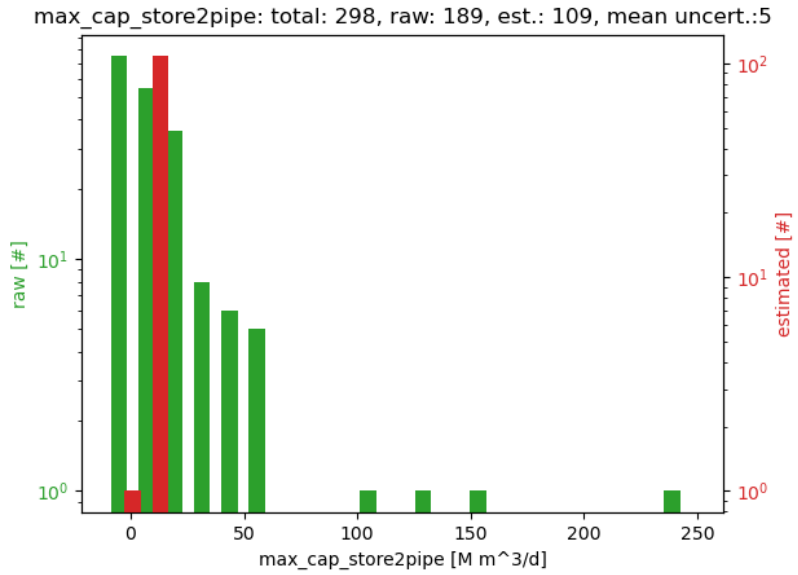


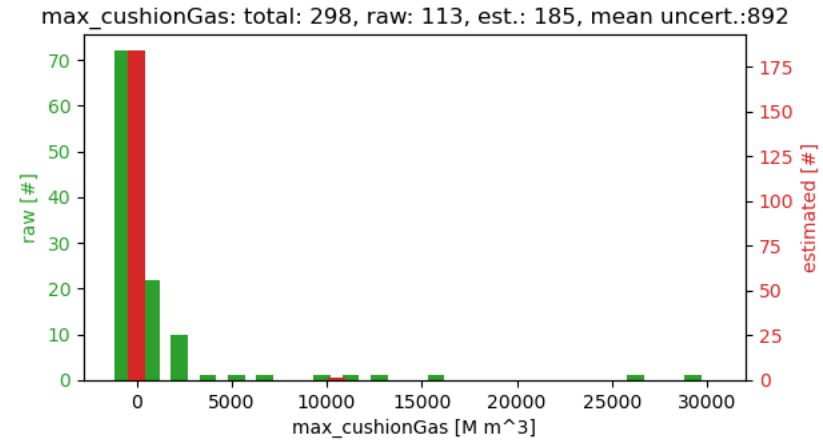
Median(max_cap_pipe2store_M_m3_per_d): 118 (mean uncert.: 11)
Lasso(max_cap_store2pipe_M_m3_per_d, Pipe_max_cap_M_m3_per_d): 1 (mean uncert.: 3)
min(raw):1.09, max(raw):244.84, min(est):2.24, max(est):7.28



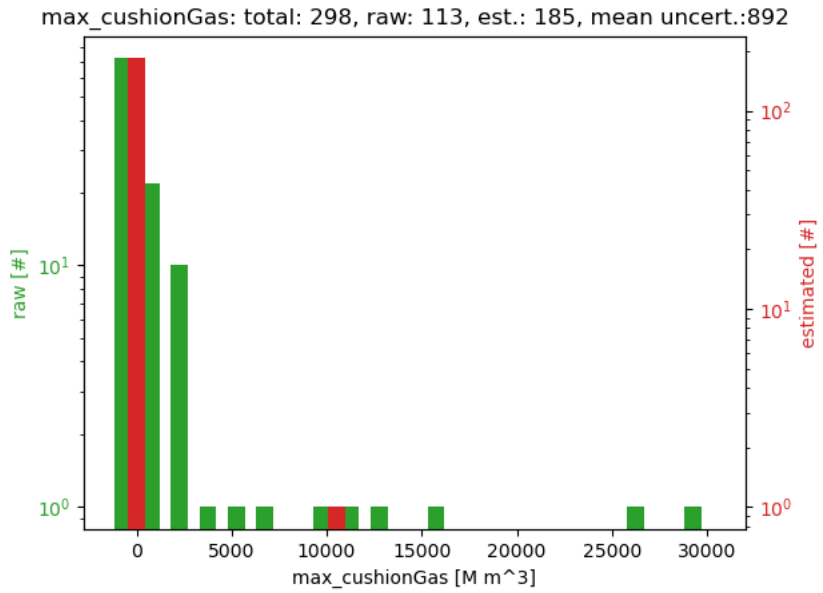


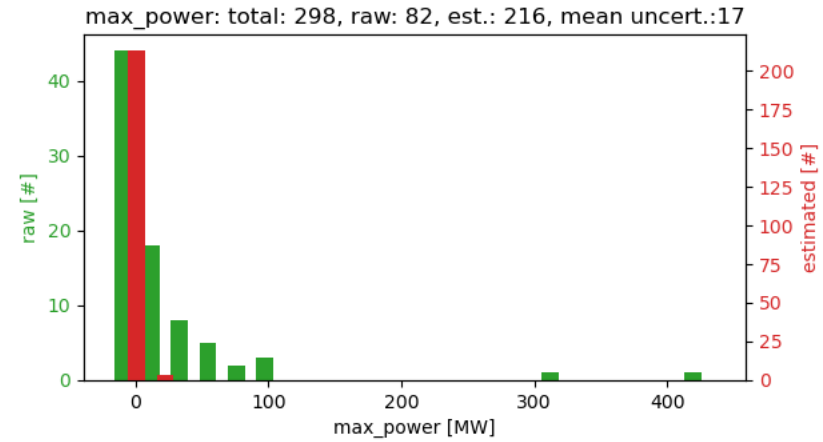
Median(max_cap_store2pipe_M_m3_per_d): 104 (mean uncert.: 13)
Lasso(max_cap_pipe2store_M_m3_per_d, max_workingGas_M_m3): 1 (mean uncert.: 6)
Lasso(max_workingGas_M_m3, Pipe_max_pressure_bar): 4 (mean uncert.: 6)
min(raw):1.01, max(raw):244.84, min(est):4.36, max(est):10.84



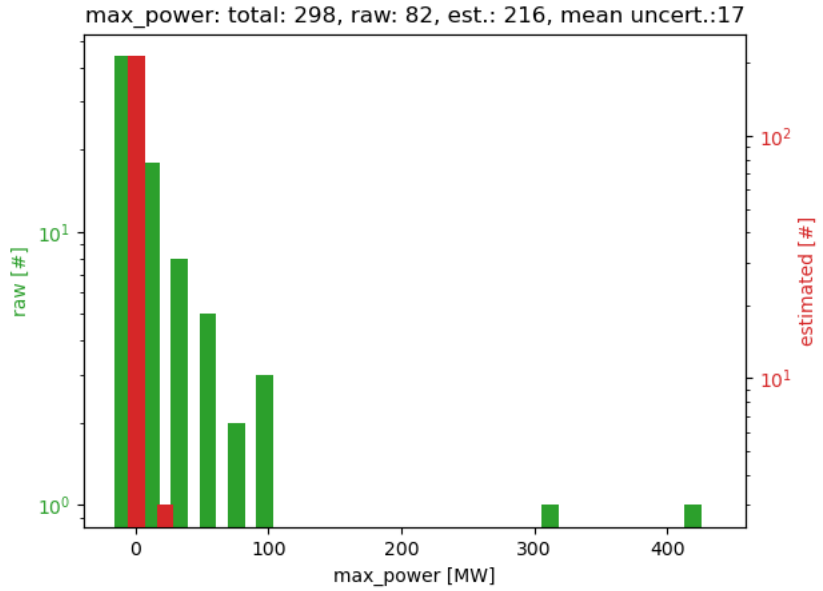


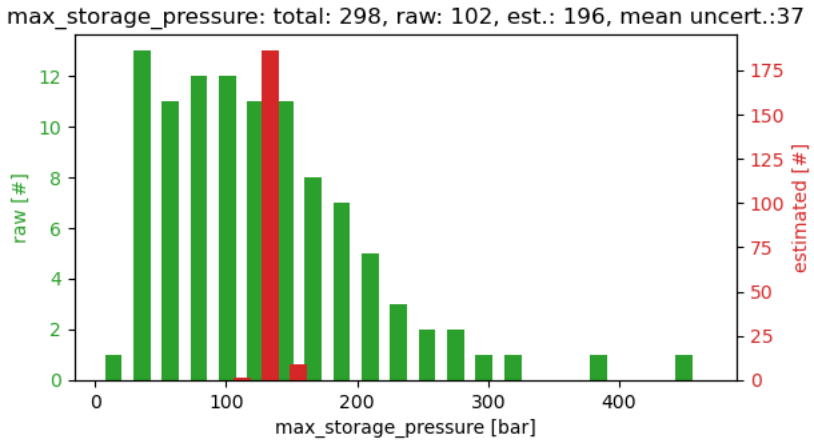
Median(max_cushionGas_M_m3): 165 (mean uncert.: 1595)
Lasso(num_storage_wells, Pipe_max_pressure_bar): 20 (mean uncert.: 134)
min(raw):1.8, max(raw):30000.0, min(est):92.29, max(est):10274.9



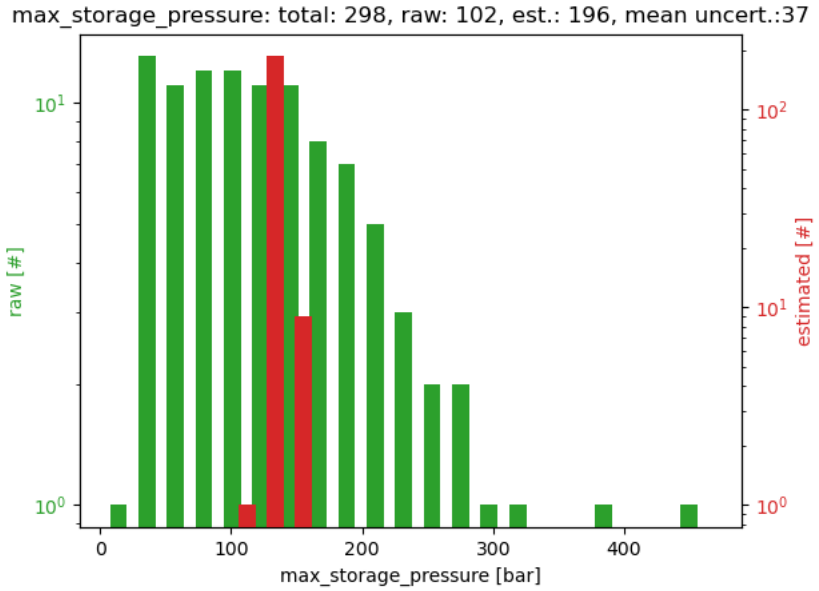


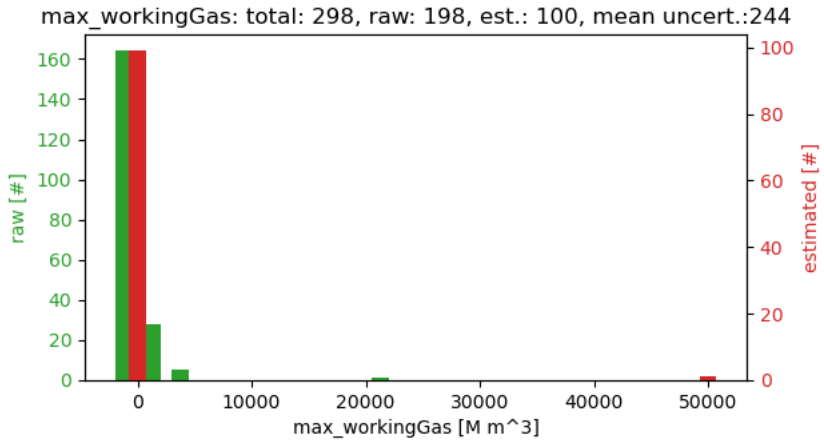
Median(max_power_MW): 198 (mean uncert.: 24)
Lasso(num_storage_wells, Pipe_max_cap_M_m3_per_d): 18 (mean uncert.: 6)
min(raw):1.0, max(raw):430.0, min(est):6.78, max(est):19.96



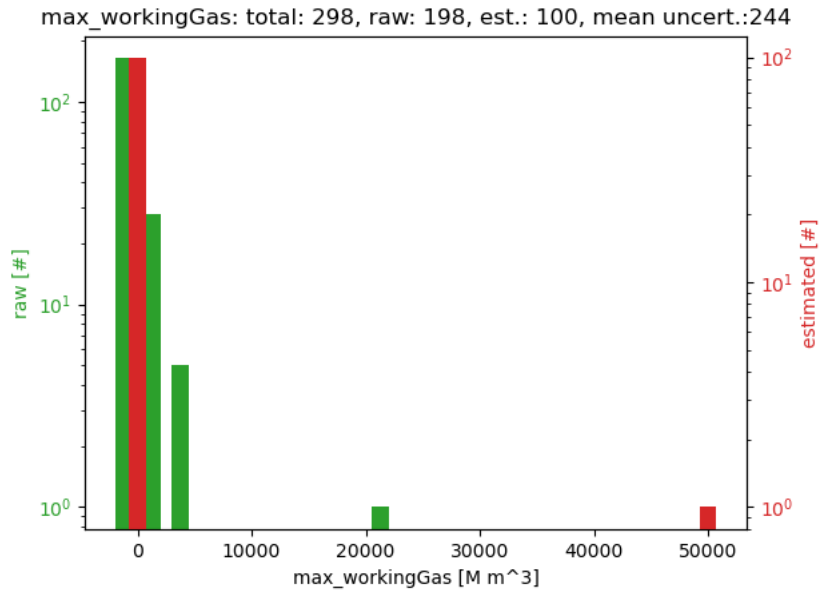


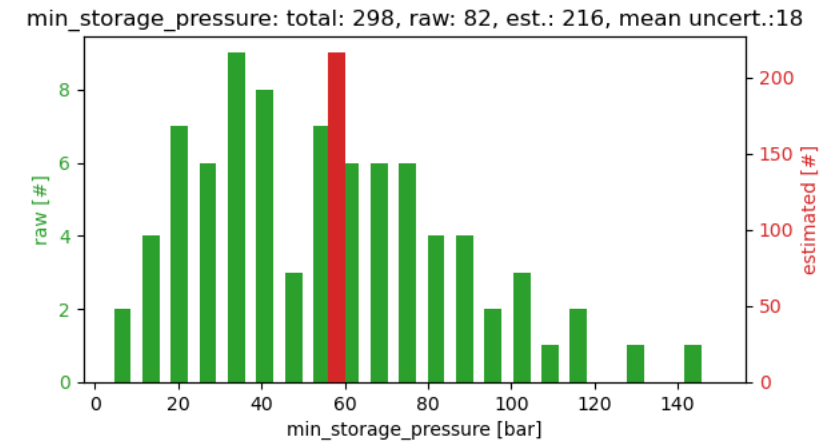
Median(max_storage_pressure_bar): 172 (mean uncert.: 59)
Lasso(min_storage_pressure_bar, Pipe_diameter_mm): 24 (mean uncert.: 39)
min(raw):25.0, max(raw):460.0, min(est):121.94, max(est):155.63



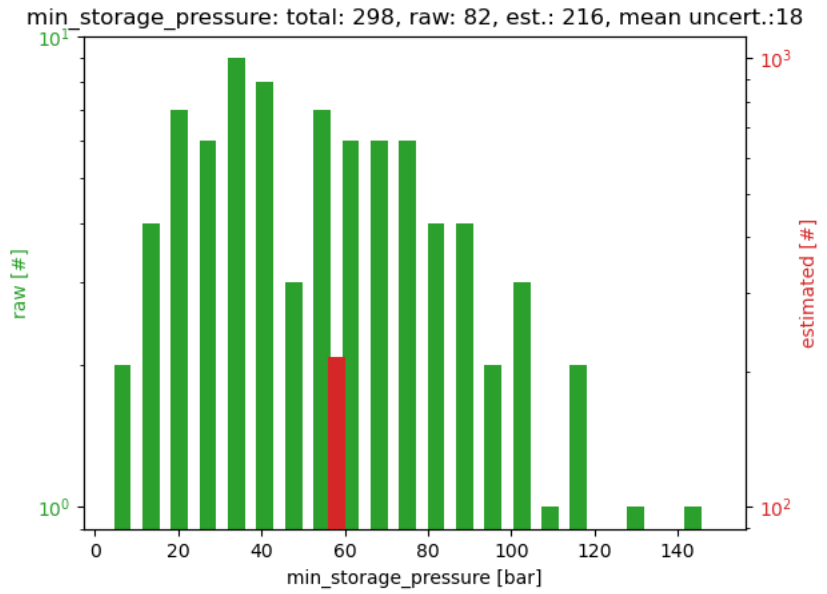


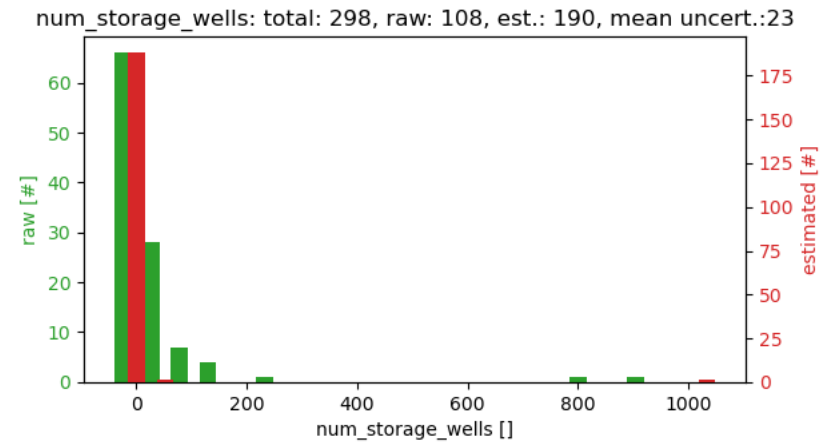
Median(max_workingGas_M_m3): 96 (mean uncert.: 649)
Squared(num_storage_wells, Pipe_max_pressure_bar): 3 (mean uncert.: 121)
M_Heuristics.adjust_AttribValues(): 1 (mean uncert.: 10000)
min(raw):11.67, max(raw):23000.0, min(est):170.19, max(est):50000.0



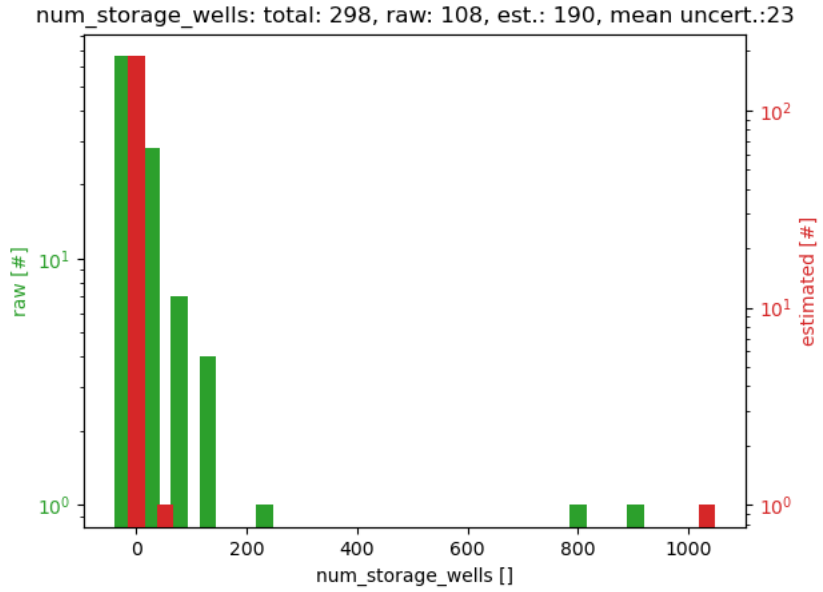


Median(min_storage_pressure_bar): 216 (mean uncert.: 25)
min(raw):10.0, max(raw):147.0, min(est):60.0, max(est):60.0





Median(num_storage_wells): 172 (mean uncert.: 40)
Lasso(max_workingGas_M_m3, Pipe_max_pressure_bar): 17 (mean uncert.: 6)
Squared(max_cap_store2pipe_M_m3_per_d, Pipe_max_pressure_bar): 1 (mean uncert.: 7)
min(raw):1, max(raw):941, min(est):9.84, max(est):1033.88



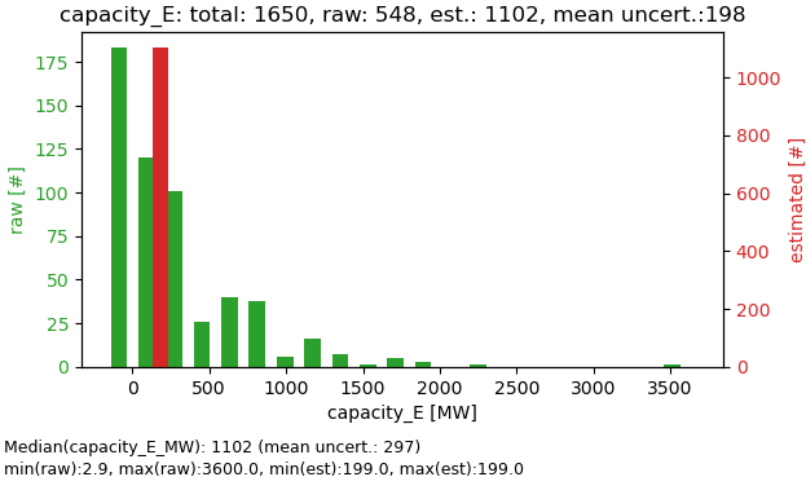
10.8.6 *Consumers*

As there was no data generated through the heuristic processes for the elements of *Consumers*, there is no need to compare attribute distributions.

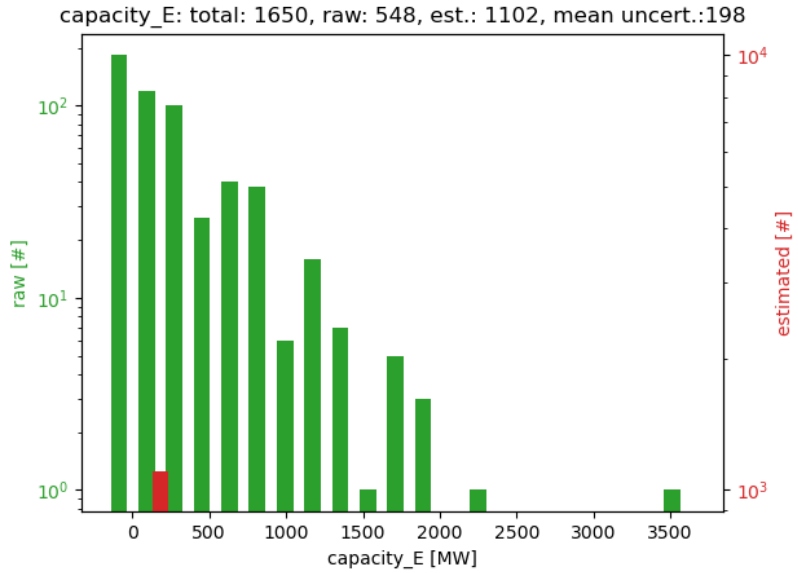
10.8.7 *PowerPlants*

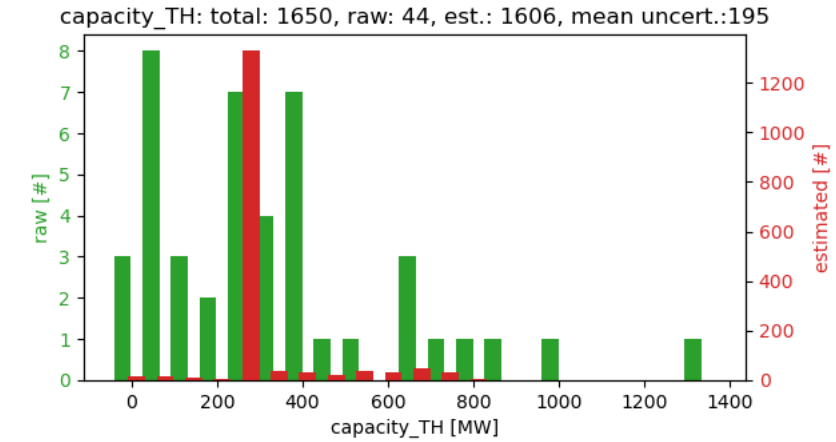
Below are the heuristic histogram plots of the component *PowerPlants* for the attributes:

- *capacity_E_MW*
- *capacity_TH_MW*

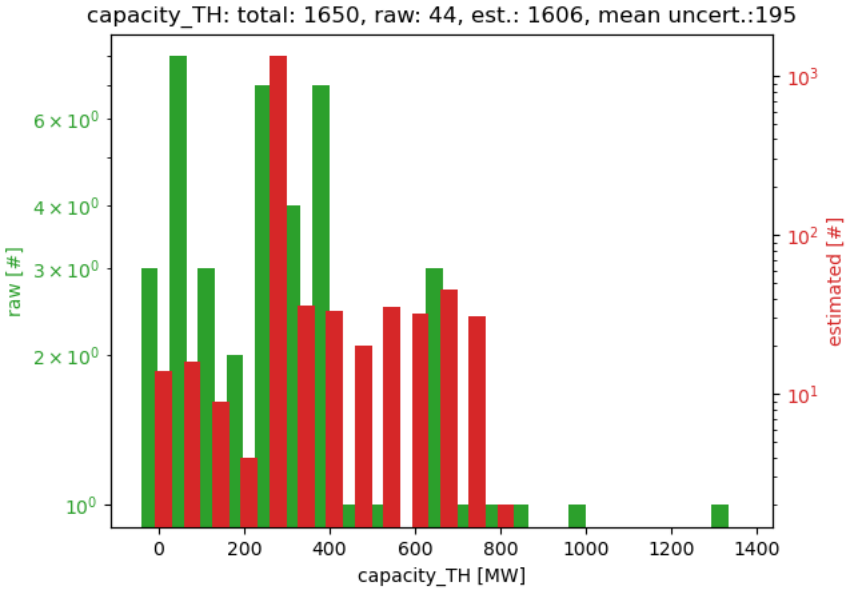


Median(capacity_E_MW): 1102 (mean uncert.: 297)
min(raw):2.9, max(raw):3600.0, min(est):199.0, max(est):199.0





Median(capacity_TH_MW): 1253 (mean uncert.: 214)
Lasso(Pipe_diameter_mm, Pipe_max_cap_M_m3_per_d): 339 (mean uncert.: 117)
M_Heuristics.adjust_AttribValues(): 14 (mean uncert.: 1000)
min(raw):25.0, max(raw):1347.0, min(est):12.0, max(est):820.73



10.9 Statistical background

Here some of the statistical methods mentioned in the document are described briefly. This is done so that actions described in this document can be understood better by the user, and is not thought to give a full explanation. Most descriptions have been copied from the Wikipedia pages, or other internet pages, and will be referenced accordingly.

10.9.1 Out-of-bag

This is a method of measuring the prediction error of random forests, boosted decision trees, and other machine learning models utilizing bootstrap aggregating (bagging) to sub-sample data samples used for training. [Wik20f]

10.9.2 Leave p-out cross-validation

The following has been copied from [Wik20b]:

"Leave-p-out cross-validation (LpO CV) involves using p observations as the validation set and the remaining observations as the training set. This is repeated on all ways to cut the original sample on a validation set of p observations and a training set.

LpO cross-validation requires training and validating the model C_p^n times, where *n* is the number of observations in the original sample, and where C_p^n is the binomial coefficient. For *p > 1* and for even moderately large *n*, LpO CV can become computationally infeasible. For example, with *n = 100* and *p = 30 %* of *100*, $C_{30}^{100} \approx 3 \times 10^{25}$."

10.9.3 Leave one-out cross-validation

The following has been copied from [Wik20b]:

"Leave-one-out cross-validation (LOOCV) is a particular case of leave-p-out cross-validation with *p = 1*.

The process looks similar to jackknife; however, with cross-validation one computes a statistic on the left-out sample(*s*), while with jackknifing one computes a statistic from the kept samples only.

LOO cross-validation requires less computation time than LpO cross-validation because there are only $C_1^n = \mathrm{n}$ passes rather than C_k^n. However, *n* passes may still require quite a large computation time, in which case other approaches such as k-fold cross validation may be more appropriate."

10.9.4 Jackknifing

The following text has been copied from [Wik20c]:

"In statistics, the jackknife is a resampling technique especially useful for variance and bias estimation. The jackknife pre-dates other common resampling methods such as the bootstrap. The jackknife estimator of a parameter is found by systematically leaving out each observation from a data set and calculating the estimate and then finding the average of these calculations. Given a sample of size *n* , the jackknife estimate is found by aggregating the estimates of each

The jackknife technique was developed by Maurice Quenouille (1924-1973) from 1949, and refined in 1956. John Tukey expanded on the technique in 1958 and proposed the name "jackknife" since, like a physical jack-knife (a compact folding knife), it is a rough-and-ready tool that can improvise a solution for a variety of problems even though specific problems may be more efficiently solved with a purpose-designed tool.

The jackknife is a linear approximation of the bootstrap."

10.9.5 Bootstrap

The following has been copied from [Wik20a]:

"Bootstrapping is any test or metric that uses random sampling with replacement, and falls under the broader class of resampling methods. Bootstrapping assigns measures of accuracy (bias, variance, confidence intervals, prediction error etc.) to sample estimates. This technique allows estimation of the sampling distribution of almost any statistic using random sampling methods.

Bootstrapping estimates the properties of an estimator (such as its variance) by measuring those properties when sampling from an approximating distribution. One standard choice for an approximating distribution is the empirical distribution function of the observed data. In the case where a set of observations can be assumed to be from an independent and identically distributed population, this can be implemented by constructing a number of resamples with replacement, of the observed data set (and of equal size to the observed data set)."

10.9.6 Z-score

The following has been summaries from [UoO14]. Two data set distribution can be compared to see if they are the same or different. The **Z**-score calculates the difference fo the two sample means in units of sample mean error.

$$Z = \frac{(X_1 - X_2)}{\sqrt{\sigma_1^2 - \sigma_2^2}}$$

with σ_1 and σ_2 being the standard error of the means of the first and second distribution, and X_1 and X_2 are the mean values of the distribution.

Here caution is to be addressed, that if the first data set is a distribution of random values, or any other data set, and the second distribution consists of data that contains only mean values of the first data set, and hence will have a standard error of the mean of zero, than the **Z**-score will be zero, as $Z \sim (X_1 - X_2)$, and as both data sets have the same mean, the **Z**-score will be zero.

So what do different **Z**-score mean? In general, in more qualitative terms:

- If the Z-statistic is less than 2, the two samples are the same
- If the Z-statistic is between 2.0 and 2.5, the two samples are marginally different
- If the Z-statistic is between 2.5 and 3.0, the two samples are significantly different
- If the Z-statistic is more then 3.0, the two samples are highly signficantly different.

10.10 Acknowledgement

We acknowledge the contribution of Dr. Ontje Luensdorf from the German Aerospace Center (DLR), Institute for Networked Energy Systems to the SciGRID_gas project.

BIBLIOGRAPHY

[AFW14] M. Ahmed, B.T. Fasy, and C. Wenk. *New Techniques in Road Network Comparison*. Penguin Random House, New York, NY, 2014.

[AG99] H. Alt and L. Guibas. *Discrete geometric shapes: matching, interpolation, and approximation-a survey*. Sack JR, Urrutia J, Handbook of Computational Geometry, Elsevier, New York, NY, 1999.

[CCCS21] 14 06 2018. [Online] Copernicus Climate Change Service. Era5 hourly data on single levels from 1979 to present. https://cds.climate.copernicus.eu/cdsapp#!/dataset/reanalysis-era5-single-levels?tab=overview, 2021. Accessed: 2021-01-15.

[Die21] J.C. Diettrich. Generation of a non-osm scigrid_gas gas transmission network data set. gitHub, 2021. to be published as part of the code release.

[DPDi20] J.C. Diettrich, A. Pluta, J. Dasenbrock, and W. i. *SciGRID_gas: The combined IGGIN gas transmission network data set*. DLR-Institut für Vernetzte Energiesysteme e.V., Oldenburg, Germany, 2020. URL: https://zenodo.org/record/4288459#.YG7aFj9CSUk, doi:10.5281/zenodo.4288458.

[DPi20] J.C. Diettrich, A. Pluta, and W. i. *SciGRID_gas: The INET gas transmission network data set*. DLR-Institut für Vernetzte Energiesysteme e.V., Oldenburg, Germany, Aug 2020. URL: https://doi.org/10.5281/zenodo.4008975, doi:10.5281/zenodo.4008975.

[DPM20a] J.C. Diettrich, A. Pluta, and W. Medjroubi. *SciGRID_gas: The combined IGG gas transmission network data set*. DLR-Institut für Vernetzte Energiesysteme e.V., Oldenburg, Germany, Aug 2020. URL: https://doi.org/10.5281/zenodo.4009129, doi:10.5281/zenodo.4009129.

[DPM20b] J.C. Diettrich, A. Pluta, and W. Medjroubi. *SciGRID_gas: The combined IGGI gas transmission network data set*. DLR-Institut für Vernetzte Energiesysteme e.V., Oldenburg, Germany, Aug 2020. URL: https://zenodo.org/record/4288468#.YG7aBz9CSUk, doi:10.5281/zenodo.4288467.

[DPM20c] J.C. Diettrich, A. Pluta, and W. Medjroubi. *SciGRID_gas: The combined IGGIELGN gas transmission network data set*. DLR-Institut für Vernetzte Energiesysteme e.V., Oldenburg, Germany, 2020. URL: https://zenodo.org/record/4642569#.YG7ZhT9CSUk, doi:10.5281/zenodo.4642568.

[DPM20d] J.C. Diettrich, A. Pluta, and W. Medjroubi. *SciGRID_gas: The combined IGGINL gas transmission network data set*. DLR-Institut für Vernetzte Energiesysteme e.V., Oldenburg, Germany, 2020. URL: https://zenodo.org/record/4288440#.YG7aHz9CSUk, doi:10.5281/zenodo.4288439.

[DPM20e] J.C. Diettrich, A. Pluta, and W. Medjroubi. *SciGRID_gas: The raw INET data set*. DLR-Institut für Vernetzte Energiesysteme e.V., Oldenburg, Germany, Aug 2020. URL: https://doi.org/10.5281/zenodo.3985249, doi:10.5281/zenodo.3985249.

[DPM20f] J.C. Diettrich, A. Pluta, and W. Medjroubi. *SciGRID_gas: The raw LKD data set*. DLR-Institut für Vernetzte Energiesysteme e.V., Oldenburg, Germany, Aug 2020. URL: https://doi.org/10.5281/zenodo.3985271, doi:10.5281/zenodo.3985271.

[DPS+21] J.C. Diettrich, A. Pluta, J.E. Sandoval, J. Dasenbrock, and W. Medjroubi. *SciGRID_gas: The combined IGGIELGNC-3 gas transmission network data set*. DLR-Institut für Vernetzte Energiesysteme e.V., Oldenburg, Germany, 2021. URL: https://zenodo.org/record/4922530#.YNLX7kxCS60, doi:10.5281/zenodo.4922529.

[Ent17] EntsoG. North West GRIP, Min Report. https://www.entsog.eu/sites/default/files/files-old-website/publications/GRIPs/2017/entsog_GRIP_NW_2017_main_xs.pdf, 2017. Accessed: 2021-03-17.

[Eur21a] EuroStat. Average size of dwelling by household type and degree of urbanisation. https://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=ilc_hcmh02&lang=en, 2021. Accessed: 2021-02-08.

[Eur21b] EuroStat. Complete energy balances. https://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=nrg_bal_c&lang=en, 2021. Accessed: 2021-02-08.

[Eur21c] EuroStat. Disaggregated final energy consumption in households - quantities. https://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=nrg_d_hhq&lang=en, 2021. Accessed: 2021-02-07.

[Eur21d] EuroStat. Employment by age, economic activity and nuts 2 regions (nace rev. 2). https://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=lfst_r_lfe2en2&lang=en, 2021. Accessed: 2021-02-08.

[Eur21e] EuroStat. Gross domestic product (gdp) at current market prices by nuts 3 regions. https://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=nama_10r_3gdp&lang=en, 2021. Accessed: 2021-03-31.

[Eur21f] EuroStat. Number of households by degree of urbanisation and nuts 2 regions. https://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=lfst_r_lfsd2hh&lang=en, 2021. Accessed: 2021-02-08.

[Eur21g] EuroStat. Population on 1 january by age group, sex and nuts 3 region. https://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=demo_r_pjangrp3&lang=en, 2021. Accessed: 2021-01-10.

[Eur21h] EuroStat. Sbs data by nuts 2 regions and nace rev. 2 (from 2008 onwards). https://appsso.eurostat.ec.europa.eu/nui/show.do?dataset=sbs_r_nuts06_r2&lang=en, 2021. Accessed: 2021-03-16.

[Gas19] GasTerra. Aardgas in Nederland. https://www.gasterra.nl/uploads/fckconnector/20d5b75a-2ad6-529e-b024-1f5bd43bbd5e, 2019. Accessed: 2021-03-17.

[GGB+20] F. Gotzens, B. Gillessen, S. Burges, W. Hennings, J. Müller-Kirchenbauer, S Seim, P. Verwiebe, T. Schmid, F. Jetter, and T. Limme. *Harmonization and development of methods for a spatial and temporal resolution of energy demands (DemandRegio)*. Forschungsstelle für Energiewirtschaft e.V., 2020. Accessed: 2021-05-18.

[Hel18] D. Helle. OpenStreetMap - Deutschland. https://www.openstreetmap.de/, 2018. Accessed: 2019-12-12.

[Kha13] Y. Khalid. What is Pickle in python? https://pythontips.com/2013/08/02/what-is-pickle-in-python/, 2013. Accessed: 2019-10-10.

[KingSpalding18] King&Spalding. LNG in Europe 2018: An Overview of LNG Import Terminals in Europe. https://globallnghub.com/wp-content/uploads/2018/09/King.pdf, 2018. Accessed: 2018-09-01.

[KKS+17] F. Kunz, M. Kendziorski, W.-P. Schill, J. Weibezahn, J. Zepter, C. von Hirschhausen, and P. Hauser. *Electricity, Heat, and Gas Sector Data for Modeling the German System*. Deutsches Institut für Wirtschaftsforschung, Daten Dokumentation 92, Berlin, 2017.

[LSS+19] P. Lustenberger, F. Schumacher, M. Spada, P. Burgherr, and B. Stojadinovic. Assessing the performance of the european natural gas network for selected supply disruption scenarios using open-source information. *Energies*, 12(4685):1–28, 2019. doi:{10.3390/en12244685}.

[MMK16] C. Matke, W. Medjroubi, and D. Kleinhans. SciGRID - An Open Source Reference Model for the European Transmission Network (v0.2). https://power.scigrid.de, 2016. Accessed: 2019-09-09.

[Mic20] Microsoft. Bing maps api. https://www.microsoft.com/en-us/maps/licensing, 2020. Accessed: 2018 to 2021.

[Nis20] A. Nisbet. Open topo data api. https://www.opentopodata.org, 2020. Accessed: 2020 to 2021.

[San21] Javier Enrique Sandoval. Estimation and simulation of a gas demand time series for the european nuts 3 regions. Master's thesis, Carl von Ossietzky Universität Oldenburg, Germany, Fak. 5, Institute of Physics (PPRE) D-26111 Oldenburg, Germany, 6 2021. Supervised by Prof. Dr. Carsten Agert, Dr. Herena Torio and Dr. Wided Medjroubi.

[San19] B. Sandvik. World Borders. http://thematicmapping.org/downloads/world_borders.php, 2019. Accessed: 2019-07-07.

[SAB+17] M. Schmidt, D. Aßmann, R. Burlacu, J. Humpola, I. Joormann, N. Kanelakis, T. Koch, D. Oucherif, M.E. Pfetsch, L. Schewe, R. Schwarz, and M. Sirvent. *GasLib—A Library of Gas Network Instances*. 2017. doi:{10.3390/data2040040}.

[sl19] scikit-learn. 1.1. Linear Models (scikit learn). https://scikit-learn.org/stable/modules/linear_model.html, 2019. Accessed: 2019-08-08.

[UoO14] USA University of Oregon. Comparing distributions: Z Test. http://homework.uoregon.edu/pub/class/es202/ztest.html, 2014. Accessed: 2020-07-07.

[Wik20a] Wikipedia. Bootstrapping (statistics). https://en.wikipedia.org/wiki/Bootstrapping_(statistics), 2020. Accessed: 2019-06-06.

[Wik20b] Wikipedia. Cross-validation (statistics). https://en.wikipedia.org/wiki/Cross-validation_(statistics)#Exhaustive_cross-validation, 2020. Accessed: 2019-07-07.

[Wik20c] Wikipedia. Jackknife resampling. https://en.wikipedia.org/wiki/Jackknife_resampling, 2020. Accessed: 2019-08-08.

[Wik20d] Wikipedia. Lasso (statistics). https://en.wikipedia.org/wiki/Lasso_(statistics), 2020. Accessed: 2020-04-04.

[Wik20e] Wikipedia. Limited-memory BFGS. https://en.wikipedia.org/wiki/Limited-memory_BFGS, 2020. Accessed: 2020-06-06.

[Wik20f] Wikipedia. Out-of-bag error. https://en.wikipedia.org/wiki/Out-of-bag_error, 2020. Accessed: 2019-07-07.

[Wik20g] Wikipedia. Transmission system operator. https://en.wikipedia.org/wiki/Transmission_system_operator/, 2020. Accessed: 2019-09-09.

[Wik20h] Wikipedia. JAGAL. https://en.wikipedia.org/wiki/JAGAL, 2020. Accessed: 2020-01-01.

[Wik21] Wikipedia. Nomenclature of territorial units for statistics. https://en.wikipedia.org/wiki/Nomenclature_of_Territorial_Units_for_Statistics, 2021. Accessed: 2021-04-10.

[BMWi11] BMWi. Forschung für eine umweltschonende, zuverlässige und bezahlbare Energieversorgung. https://www.bmwi.de/Redaktion/DE/Publikationen/Energie/6-energieforschungsprogramm-der-bundesregierung.pdf?__blob=publicationFile&v=12, 2011. Accessed: 2019-02-02.

[BMWi20] BMWi. Home page of BMWi. https://www.bmwi.de/Navigation/DE/Home/home.html, 2020. Accessed: 2020-03-03.

[BundesregierungDeutschland20] Bundesregierung Deutschland. Home page of Bundesregierung Deutschland. https://www.bundesregierung.de/Webs/Breg/DE/Themen/Energiewende/_node.html, 2020. Accessed: 2020-01-01.

[EntsoG20] EntsoG. Home page of EntsoG. https://www.entsog.eu/, 2020. Accessed: 2020-03-03.

[GasIEurope20] Gas Infrastructure Europe. Home page of Gas Infrastructure Europe. https://agsi.gie.eu, 2020. Accessed: 2020-01-01.

[GasSEurope20] Gas Storages Europe. Home page of Gas Storages Europe. https://www.gie.eu/index.php/transparency/gse-transparency-template, 2020. Accessed: 2020-01-01.

[Gassco20a] Gassco. Data page of facilities from Gassco. https://www.npd.no/en/about-us/information-services/available-data/map-services/, 2020. Accessed: 2020-01-01.

[Gassco20b] Gassco. Home page of Gassco Norway. https://www.gassco.no/en/, 2020. Accessed: 2020-01-01.

[IGU20] IGU. Home page of International Gas Union. https://www.igu.org/, 2020. Accessed: 2018-10-01.

[nationalGrid20] nationalGrid. Home page of National Grid UK. https://www.nationalgrid.com/uk/, 2020. Accessed: 2018-10-01.