Categorical Databases for Mathematical Formalization of AC Optimal Power Flow

Masoud Barati
Electrical and Computer Engineering Department
Swanson School of Engineering, University of Pittsburgh
Pittsburgh, PA, USA
masoud.barati@pitt.edu

2023 IEEE Texas Power and Energy Conference (TPEC) | 978-1-6654-9071-9/23/$31.00 ©2023 IEEE | DOI: 10.1109/TPEC56611.2023.10078526

***Abstract*—It has been decades since category theory was applied to databases. In spite of their mathematical elegance, categorical models have traditionally had difficulty representing factual data, such as integers or strings. This paper proposes a categorical dataset for power system computational models, which is used for AC Optimal Power Flows (ACOPF). In addition, categorical databases incorporate factual data using multi-sorted algebraic theories (also known as Lawvere theories) based on the set-valued functor model. In the advanced metering infrastructure of power systems, this approach is capable of handling missing information efficiently. This methodology enables constraints and queries to employ operations on data, such as multiplicative and comparative processes, thereby facilitating the integration between conventional databases and programming languages like Julia and Python's Pypower. The demonstration illustrates how all elements of the model, including schemas, instances, and functors, can modify the schema in ACOPF instances.**

***Index Terms*—ACOPF, Category, Compositional, Database, Functor.**

I. Introduction

The power system is composed of a centralized bulk generation sector and a downstream dedicated consumption sector. In addition, the model can be adapted to include residential loads with or without distributed energy resources downstream at the end user. The users are aware of how to participate meaningfully in the grid. Figure 1 illustrates the interactions between the different sectors of the power system. There are two types of interactions that are possible in this model, physical interactions and communication interactions. It is important to note that the dotted lines represent the actual flow of power, which is similar to the power flow in a real power system. In this model, the enabling and challenging components are all solid lines, which are secure communication flows. System operators are primarily responsible for controlling these interconnected systems under coordinated platforms. An operator of a power system typically operates the optimal power flow in the day ahead and the real-time market in order to provide the least cost or maximum social welfare operation, the minimum congestion, overloading, and minimize the violation of the state variable. With the use of a

Masoud Barati is with the Department of Electrical and Computer Engineering and Industrial Engineering, Swanson Schoolof Engineering, University of Pittsburgh, PA, USA. This work was supported by the NSF ECCS Award #1711921.

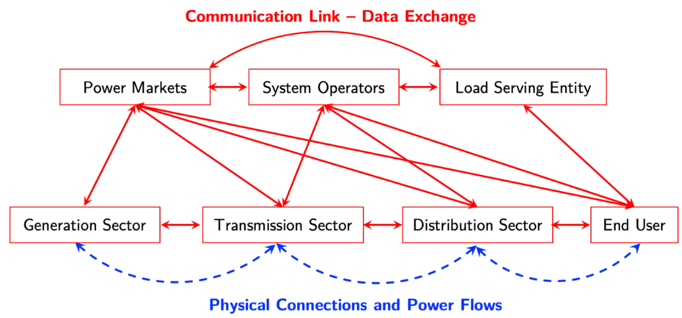

Fig. 1. Interactions among different power system sectors

cloud, the operation conditions, such as the topology, physical operation, and economic and financial characteristics of each sector, are represented. Each of these clouds can be zoomed in on. It is evident that there are many standards being developed that provide guidance, such as specifying the exchange of model information among all sectors while protecting the privacy of all participants. The purpose of this research is to investigate the information model and its exchange of information. There are many formal features of AC optimal flow (ACOPF), but the relationship between these pieces still needs to be mapped out [1], [2], [3]. This connectivity is formalized by reconstructing existing workflows and best practices in categorical terms. System integration generally requires two factors [4]: First, The first step is to understand the physics of the power system and ACOPF; Second, The remainder is a matter of mathematics [5], [6], [7].

It is recommended to represent systems as algebras if the science of interconnection in a dataset is to be described by given mathematics. Those algebras are then composed together in category theory [4].

An algebraic structure on a graph is a general definition of a category. Our goal is to provide an intuitive explanation of the fundamental concepts of the categorical theory in datasets of ACOPF. Objects are grouped into categories along with arrows (including identity arrows) which a composition rule can formally link together. Examples of valuable categories include sets and functions, graphs and graph transformations, and states and transitions between them [8], [9]. In Section II, a paradigm is presented for an issue related to ACOPF, including the definition of category theory, its underlying elements, and the conical form of ACOPF problem. An instance of a database is a functor derived from a database schema (i.e., a directed, recursive multigraph) and a database schema is a finitely introduced category. Section III describes the implementation in CATLAB [10]. As schemas are viewed as categories, model mappings are functors from schema to schema. A functor of this type will inevitably result in many translations between instances adhering to different schemas at the source and destination. As a result of these mappings, a framework for linking numerous models and technologies is described in Section IV. Section V examines a particular class of ACOPF. It is intended that the concept of different generation units in ACOPF provides a formal abstraction or characterization of the critical characteristics of a diverse range of energy resources, such as renewable energy resources and batteries. Our study presents a class of models of generation resources as instinctive graphs, where morphisms are viewed as model transformations or synthesis. A monoidal structure is permitted in this category, with different generation unit aggregation as a tensor product.

II. Problem Formulation and Model Architecture

The purpose of this section is to provide an overview of the proposed model in its essential components.

A. Category Theory

There is this common information model which is not unique to power systems, while there is a set of standards that focus on utilizing that in the context of power systems. They are basically UML-based kind of models, and then they use XML for their Model exchange. There is a whole bunch of other standards that are related in really complicated ways to all the things in Fig. 1. Therefore, our approach is based on techniques from categorical databases. This technique applies to ACOPF problems for model management. In this model, we define some schemas for ACOPF as an object and then define a set-valued function in instances. The objects will be connected by functors between the schemas, those will induce functors between the instances. One of the main objects is the solver and many other tools, such as a schema reader for running the ACOPF problems. In the end, we embedded connecting different tools in this categorical database model [11], [12]. The following diagram is meant to represent data in a category consisting of three objects X, Y, and Z, identity morphisms on theirs $id's$, and two morphisms f and g, along with their composition rule $g \circ f$. Figure 2 shows a simple category and its morphisms [13], [14], [15].

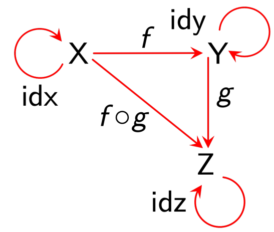

Fig. 2. A graphical representation of a category is provided, with objects X, Y, and Z, as well as morphisms f, g, and $g \circ f$. Additionally, the category includes three identity morphisms, namely idx, idy, and idz.

B. Categorical database as an object and functor design

There is this standard information model which is not unique to power systems, while there is a set of standards that focus on utilizing that in the context of power systems. They are basically UML-based kind of models, and then they use XML for their Model exchange. There is a whole bunch of other standards that are related in really complicated ways to all the things in Fig. 1. Therefore, our approach is based on techniques from categorical databases. This technique applies to ACOPF problems for model management. In this model, we define some schemas for ACOPF as an object and then define set-valued functions in instances based on Lawvere's theory [11]. The objects will be connected by functors between the schemas, and those will induce functors between the instances. One of the main objects is the solver and many other tools, such as a schema reader for running the ACOPF problems. In the end, we embedded connecting different tools in this categorical database model. The first step in modeling the behavior of the system is to create a Markov chain. As a result, different operating points indicate states which correspond to various failure modes, working models, dispatching, or other changes to the ACOPF problem to which we would like to assign transition probabilities. We implemented our proposed model using the CATLAB tool. The beauty of this model is that we have some groups as an object for the dataset, and we spent a lot of time trying to build these realistic data models for all generation resources and any other system assets. In order to build a base model, you must define functors between these objects at the instance level. A base model can then be constructed from the combination of these functors and applied to other instances and case studies. As a result, it is similar to building up a model repository of these asset models under various conditions. It helps the system operator utilize this model as a generic tool for operating conditions. For instance, running the ACOPF for various use cases and test cases at steady-state operation.

C. AC Optimal Power Flow Model

Physical power flow mapping constitutes the fundamental basis for traditional grid monitoring and planning instruments, as demonstrated in [5],

$$\begin{aligned} p_i &= \sum_{k=1}^{n} |v_i|\,|v_k|\left(g_{ik}\cos\theta_{ik} + b_{ik}\sin\theta_{ik}\right), \\ q_i &= \sum_{k=1}^{n} |v_i|\,|v_k|\left(g_{ik}\sin\theta_{ik} - b_{ik}\cos\theta_{ik}\right), \end{aligned} \tag{1}$$

The expression under consideration holds for indices $i = 1, \cdots, n$. The variables p_i and q_i are used to denote the real and reactive power injections at the i^{th} bus, respectively. The $(i,k)^{th}$ element in the admittance matrix, denoted by $Y = G+\sqrt{-1}B$, is represented as $\left(g_{ik}+\sqrt{-1}b_{ik}\right)$. The magnitude of voltage at the i^{th} bus is represented as $|v_i|$, and the phase angle difference between the i^{th} and k^{th} bus is represented by θ_{ik}.

To create a model for the optimization solver in the following content, we transform the trigonometric functions into polynomial functions using rectangular coordinates [1], [6], [7]. This is accomplished by the following definition:

$$u_i = |v_i| \cos\theta_i, \quad w_i = |v_i| \sin\theta_i, \tag{2}$$

$$v_i = u_i + \sqrt{-1}w_i \tag{3}$$

where u_i and w_i are the real and imaginary parts of the bus voltage phasor, the nodal power flow equations of the power network (1) could be represented as functions of u_i and w_i :

$$\begin{aligned} p_i &= \sum_{k=1}^{n}\left(u_i u_k g_{ik} + w_i w_k g_{ik} + w_i u_k b_{ik} - u_i w_k b_{ik}\right) \\ q_i &= \sum_{k=1}^{n}\left(w_i u_k g_{ik} - u_i w_k g_{ik} - u_i u_k b_{ik} - w_i w_k b_{ik}\right) \end{aligned} \tag{4}$$

In addition, we define

$$\boldsymbol{u} = [u_1; \cdots; u_n], \quad \boldsymbol{w} = [w_1; \cdots; w_n], \quad \boldsymbol{x} = [\boldsymbol{u}; \boldsymbol{w}]. \tag{5}$$

It is then possible to abstractly represent the underlying power flow equations (5) in the following way:

$$\begin{aligned} p_i &= f_{p_i}(\boldsymbol{x}), \\ q_i &= f_{q_i}(\boldsymbol{x}). \end{aligned} \tag{6}$$

According to traditional practices, the power flow equations f_{p_i} and f_{q_i} are governed by the network topology and electrical parameters of the transmission lines, transformers, shunt, and series components of the system. Note that all parameters and topology of the system are assumed to be known and provided. A lack of accurate topology information and missing line parameters may prevent the use of physical law-based illustrations of power flow equations. Increasing data availability in distribution grids has been observed as a solution to the problem. The AC power flow problem originally is non-convex and in the form of SDP, the $X_{ii} = u_i w_i$ and $X_{ij} = u_i w_j$ are the hermitian matrix entries in (7).

$$\begin{aligned} vv^{\mathrm{T}} &= \begin{pmatrix} |u_1|^2 & u_1 w_1 & \cdots & u_1 w_n \\ u_2 w_1 & |u_2|^2 & \cdots & u_2 w_n \\ \vdots & \vdots & \ddots & \vdots \\ u_n w_1 & u_n w_2 & \cdots & |u_n|^2 \end{pmatrix} \\ &= \begin{pmatrix} X_{11} & X_{12} & \cdots & X_{1n} \\ X_{21} & X_{22} & \cdots & X_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n1} & X_{n2} & \cdots & X_{nn} \end{pmatrix} = X. \end{aligned} \tag{7}$$

In the SOCP formulation, the following additional equations are added to the ACOPF (4) as follows,

$$X_{ii}X_{jj} \geq X_{ij}^2 \tag{8}$$

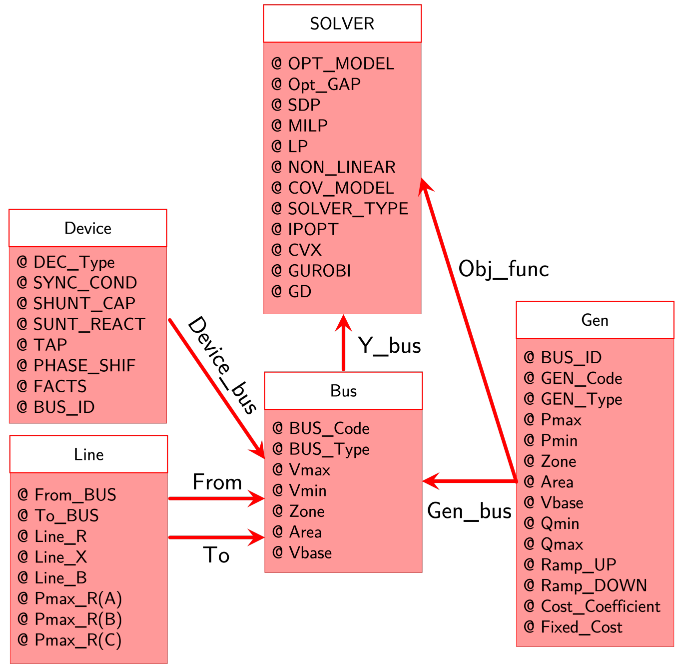

Fig. 3. creation of a schema for ACOPF problem in CATLAB based on Pypower

III. Schema Creation with CATLAB

As a result of the ACOPF problem, we can determine how hard it is for active and reactive power to flow through the network in an optimal and feasible manner. In this type of diagram, it is much more complicated than we can imagine. Objects or entities of the ACOPF schema are shown in this diagram. The ACOPF (or even a customized ACOPF with different objective functions and constraints) can have all the attributes required to specify one of these problems in the physical constraint setup or in the engineering information setup. For example, the maximum voltage across the transmission line, the maximum current through the line, or some sort of type system on any bus in the system.

A. Schema for ACOPF datasets

Pypower, a Python implementation, was used to create these objects and attributes [16]. A brief description of some of the attributes can be found in Figure 3. These include Devices, line datasets, bus datasets, generation datasets, and optimization solver attributes. We then constructed a QL or CQL for this schema based on all of Pypower's test cases.

Constraints can be enforced using path equations in CQL. As an example, in Pypower, buses have an `integer_ID`, `BUS_ID`, and branches have integer variables, `From_BUS`, `To_BUS`. We have the same statement for the generation unit at each network bus. Figure 3 shows the category of buses, lines, and generation unit connectivity in the physical network. These variables specify the values of `BUS_ID` for the source and target of each branch. Our schema on the left reinforces this constraint with the equations on the right,

```
FROM_Bus=s.BUS_ID,
TO_Bus=t.BUS_ID,
Bus_Gen_ID_Bus=TO.BUS_ID.
```
(9)

The Pypower and Gurobi solvers provide numerical solutions to AC optimal power flow problems [16].

B. Definition of functors among ACOPF's datasets

A fundamental question that arises at this stage is how to define the mapping, represented as a functor, between the corresponding schemas in the datasets of ACOPF. These functors enable the conversion of a model of an abstract solver into a model of a specific solver. An instance of such a functor is the inclusion of entities and attributes from the generic model, with a selection of attributes from the particular model remaining unaffected.

It is proposed that a functor, denoted as F, can be defined to map between two databases, S representing line data and T representing bus data. In [17], it is established that functors $F : S \to T$ lead to the formation of triples, consisting of $\Sigma_F, \Delta_F, \Pi_F$, among the instances affiliated with these functors. Here, Δ_F maps from T-Inst to S-Inst and Σ_F, Π_F map from S-Inst to T-Inst. As a result of these functors, it is possible to formulate queries between databases, with $Q : S \to T$, as described in [18]. Evaluating such a query is possible via $\epsilon(Q)$, which maps from S-Inst to T-Inst, in a manner analogous to relational data models. Additionally, the functorial data model facilitates the dual operation of co-evaluation, represented by $\iota(Q)$, which maps from T-Inst to S-Inst. With the combination of functors for data migration and queries, CQL presents a viable solution for transforming data from various solvers, given its capability to compute colimits within categories of instances.

The objective of this paper is to outline the method for transforming instances from one solver schema S to another solver schema S' through the use of a generic solver schema G. The transformation from S to G is represented by $Q : S \to G$, while the transformation from S' to G is denoted by $Q' : S' \to G$. The combination of these transformations, $(Q) \circ \iota(Q') : S\text{-Inst} \to S'\text{-Inst}$, translates the generic data from S to S', but can result in loss of non-generic data linked to the instance. To mitigate this loss, an auxiliary schema A is introduced. The functors $F : A \to S$ and $F' : A \to S'$ integrate the data of A into the two solver schemas. The composite functor $\Delta_F \circ \Sigma_{F'} : S\text{-Inst} \to S'\text{-Inst}$ transfers data from S to A and then to S'. The resulting instances in S' are represented by $S' : \iota(Q')(\epsilon(Q)(I))$ and $\Sigma_{F'}(\Delta_F(I))$, which capture the original instance I in G and the data in A, respectively. By employing an appropriate Colimit in S'-Inst, a single instance of S' can be derived, incorporating all the translated data from S.

IV. Simulation Results

In this section, we illustrate the proposed categorical dataset for ACOPF. To examine the efficiency of the proposed model, we sequentially added some datasets as new objects into the model to evaluate the performance of category theory as a formalism to the ACOPF problem. The physical model of the instances carried out by Pypower and the optimization problems handled by Gurobi solver. The ACOPF is solved by two existing models, Semidefinite Programming (SDP) and Second Order Cone Programming (SOCP) which have been proposed in [1], [2], [3]. Four case studies are investigated.

- **Case 1)** The solution results of DCOPF when the reactive power and ohmic losses have ignored.
- **Case 2)** The solution results of linearized ACOPF when the ohmic loss has ignored.
- **Case 3)** The solution results of ACOPF when the optimality gap is less than %0.05.
- **Case 4)** The solution results of ACOPF when the optimality gap is less than %0.05, and one generation is connected to bus 10 in all test cases.

The simulation results are given in Tables I, II, III, and IV. Cases 1 and 2 illustrate a simple network and a simple optimization problem. In comparison to case 3, the results of this case are very rapid. In case 4, we assumed that a new generation unit was added to the system. The bus and generation schema will be changed as a result of this event. From a system operator's perspective, it will add additional time to the realistic ACOPF total solution time. The time overhead when we do not use the categorical dataset techniques is computed in Galaxy servers at Center for Research Computing (CRC) cluster (galaxy servers are using HTC cluster), located at University of Pittsburgh [19]. The results of overhead time with categorical datasets are also computed in CATLAB.

Table IV indicates that the categorical database has less

TABLE I
The solution results of DCOPF in case 1, when the reactive power and ohmic loss has ignored.

No. of Buses	Total time (s)
57	0.12
118	0.30
300	0.48

TABLE II
The solution results of linearized ACOPF in case 2, when ohmic loss has ignored.

No. of Buses	Total time (s)
57	0.09
118	0.09
300	0.25

TABLE III
The solution results of ACOPF in case 03, when the gap is less than %0.05.

No. of Buses	No. of Iter		Total time (s)		Last iter. Time (s)	
	SDP	SOCP	SDP	SOCP	SDP	SOCP
57	6	3	1.8	1.4	0.4	0.6
118	12	5	9.8	6.2	1.3	1.8
300	9	6	50.2	36.3	8	10

TABLE IV
TIME OVERHEAD COMPARISON BETWEEN TWO CASES 3 AND 4 BY ADDING A NEW GENERATION UNIT AT BUS 10.

No. of buses	Time overhead in Case 3	Time overhead in Case 4
57	272%	5%
118	314%	2.5%
300	65%	3%

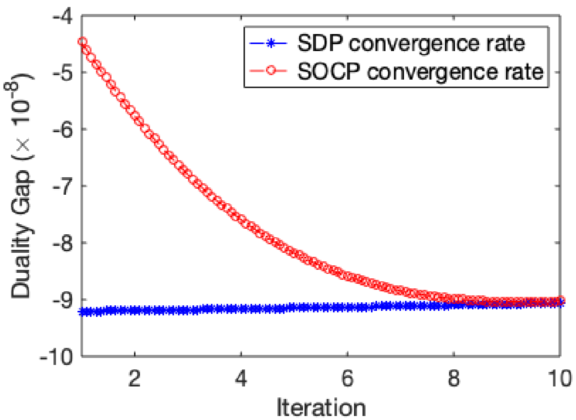

Fig. 4. Convergence graph of SDP and SOCP for large-scale IEEE 300-bus test system

overhead time in case 4 while the overhead time is very high in our simulation studies. The percentage is calculated based on the original total time without overhead time. This relative overhead appears to vary greatly depending on the instance. Those results confirm the superior performance of the categorical dataset model. As a result, any topological changes to the schema are very easy, and we can regard the schema as a standard tool. This means that we can use this structure as a pocket calculator since we have an organized set of schemas and functors at our disposal. An IEEE 300-bus test system is utilized to study the convergence rate of SDP and SOCP and the impact of the proposed database categorical method on the ACOPF problem. The value of the duality gap is equal to the $\texttt{Duality Gap} = \frac{\texttt{UB - LB}}{\texttt{LB}}$. The tolerance of the duality gap is chosen by default 10^{-7}. Figure 4 illustrates the convergence of SDP and SOCP for a large-scale IEEE 300-bus test system. The last iteration of SDP occurred at 8 seconds while the last iteration of SOCP occurred at 10 seconds.

V. CONCLUSION

We provided an overview of a new platform for computational techniques related to power systems, such as ACOPF. A category-theoretic approach to problems involving multiple models, tools, and scales was demonstrated by the framework.

We also considered a category whose objects are power system datasets and optimization solver computational layers whose morphisms were transformations between topological and parametric changes in the grid in response to natural disasters or any unexpected events such as cyber-physical attacks on power systems. To build more complex networks and systems, it would be helpful to level-shift and consider a category whose morphisms are datasets as schema. It will take additional work to extend this idea to other management layers of power systems, such as communication, gas networks, and water nexuses.

REFERENCES

[1] J. Lavaei and S. H. Low, "Zero Duality Gap in Optimal Power Flow Problem," in IEEE Trans. Power Syst., vol. 27, no. 1, pp. 92-107, 2012. D. Molzahn and I. Hiskens, "A Survey of Relaxations and Approximations of the Power Flow Equations", Found. Trends in Elec. Ener. Syst., vol. 4, no. 1-2, pp 1-221, 2019.
[2] D. Bienstock and A. Verma, "Strong NP-hardness of AC power flows feasibility," Oper. Res. Lett., vol. 47, pp. 494-501, 2019.
[3] C. Coffrin, H. Hijazi and P. van Hentenryck, "Strengthening the SDP relaxation of AC power flows with convex envelopes, bound tightening, and valid inequalities," IEEE Trans. Power Syst., vol. 32, no. 5, pp. 35493558, 2017.
[4] Soumya Kundu, Karanjit Kalsi, and Scott Backhaus. Approximating flexibility in distributed energy resources: A geometric approach. In 2018 Power Systems Computation Conference (PSCC), pages 1-7. IEEE, 2018.
[5] C. Josz, S. Fliscounakis, J. Maeght, and P. Panciatici."AC power flow data in MATPOWER and QCQP format: iTesla, RTE snapshots, and PEGASE." arXiv preprint arXiv: 1603.01533, 2016.
[6] A. Eltved, J. Dahl, and M. S. Andersen, "On the robustness and scalability of semidefinite relaxation for optimal power flow problems," Optim. Eng., vol. 21, no. 2, pp. 375-392, 2020.
[7] D. K. Molzahn, J. T. Holzer, B. C. Lesieutre and C. L. DeMarco, "Implementation of a Large-Scale Optimal Power Flow Solver Based on Semidefinite Programming," IEEE Trans. Power Syst., vol. 28, no.4, pp. 3987-3998, 2013.
[8] Serge Abiteboul, Richard Hull, and Victor Vianu. Foundations of Databases. Addison-Wesley, 1995. IsBn: 0-201-53771-0.
[9] Jiří Adámek and Jiří Rosický. Locally presentable and accessible categories. London Mathematical Society Lecture Note Series 189. Cambridge University Press, 1994. Isn: 0-521-42261-2.
[10] Online_https://algebraicjulia.github.io/Catlab.jl
[11] J. Adámek, J. Rosický, and E. M. Vitale. Algebraic theories. Cambridge Tracts in Mathematics 184. Cambridge University Press, 2011. IsbN: 978-0-52111922-1.
[12] Francis Borceux. Handbook of categorical algebra. Vol. 2: Categories and structures. 3 vols. Encyclopedia of Mathematics and its Applications 51. Cambridge University Press, 1994. IsbN: 0-521-44178-1.
[13] Michael Barr and Charles Wells. Toposes, Triples and Theories. A Series of Comprehensive Studies in Mathematics 278. Springer-Verlag, 1985. ISBN: 0-387-96115-1.
[14] AnHai Doan, Alon Halevy, and Zachary Ives. Principles of Data Integration. 1st. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2012. IsBN: 0124160441,9780124160446.
[15] Michael Fleming, Ryan Gunther, and Robert Rosebrugh. "A database of categories". In: Journal of Symbolic Computation 35.2 (Feb. 2003), pp. 127-135. ISSN: 0747-7171. DOI: 10.1016 0747-7171 (02) 00104-9.
[16] Online_https://github.com/rwl/PYPOWER
[17] David I. Spivak. Functorial data migration. Information and Computation, 217:31 - 51, 2012. ISSN 0890-5401. DOI: https://doi.org/10.1016/j.ic.2012.05.001. URL http://www.sciencedirect.com/science/article/pii/s0890540112001010.
[18] Patrick Schultz and Ryan Wisnesky. Algebraic data integration. Journal of Functional Programming, 27:e24, 2017. DOI: 10.1017/S0956796817000168.
[19] https://crc.pitt.edu/galaxy