# The Common Electric Power Transmission System Model v0.1.0-alpha

Carleton Coffrin, more authors forthcomming *

October 2019

## 1 NOTE

This is a work in progress draft!

## 2 Introduction

As the power systems research community continues to develop new methods for the design and operations of power systems, the data requirements for supporting those innovations are continually evolving. In the context of modeling electric power transmission systems, the broad adoption of the MATPOWER v2 data format [1] has resulted in a de facto standard for encoding these datasets. However, a survey of emerging power system modeling tools such as EGRET [], GRG [], Grid Optimization Competition Challenge 1 [1], IIDM [], Pandapower [2], PowerModels [3], PowerSystems [2], PyPSA [3], GridCal [4], PSAT and DOME [5] reveals that each tool provides ad hoc extensions to the core MATPOWER data requirements to accommodate their design goals. This trend suggests that there is a need to develop new standards for electric power transmission system models, which can capture more of data requirements of these emerging tools.

To help support standardization across multiple emerging software tools, this document proposes a Common Electric Power Transmission System Model (CTM), which provides specifications of network component models, parameter names, parameter units, and mathematical specifications. The CTM model is not a data format in a strict sense, but rather an abstract specification that provides guidelines for standardization of the core modeling features that span a variety of research-focused software tools. Note that this objective distinguishes CTM from data formats like CIM [], which focuses on a very comprehensive data model for information exchange across enterprise IT systems. The remainder of this document discusses the motivations, objectives, and specification of the CTM model.

Figure idea, Venn diagram showing the data models of different tools. Compare and contrast the MATPOWER format and CTM.

## 3 Scope and Objectives

The current version of CTM focuses on single-phase equivalent, quasi-steady state models of electric power transmission networks. This focus reflects the core features of the previously mentioned power system analysis tools []. With this scope in mind, the CTM specification targets the non-linear Alternating Current (AC) versions of following foundational analysis tasks,

- Power Flow (PF)

*Corresponding author cjc@lanl.gov

[1] https://gocompetition.energy.gov/

[2] https://github.com/NREL/PowerSystems.jl

[3] https://pypsa.org/

[4] https://github.com/SanPen/GridCal

[5] http://faraday1.ucd.ie/software.html

- Optimal Power Flow (OPF)
- Unit Commitment (UC)

It is important to note that a variety of analysis tasks can be defined by combinations or subset of these foundational ones. For example, Production Cost modeling can be conducted with a subset of UC parameters (double check this claim) and UC with network constraints requires the combination of the data for both the UC and OPF tasks. Note on multi-period support?

**Objectives** The overarching goal of CTM is to provides a common foundation for research on emerging trends in transmission system research. To that end, CTM was designed to meet the following objectives,

- A sufficient level of detail to provide a good approximation of industrial transmission system network models (e.g. PSSE v33)
- Support for both bus-branch and node-breaker network models
- Support for energy storage
- Support HVDC lines
- Define both input parameters and solution value parameters
- Build on the lessons learned from multiple open-source model specifications
- Strive to be cross-compatible with as many programming languages and data formats as possible
- Version control to carefully track enhancements and modifications to the CTM specification

**CTM Implementations** Software tools that adopt CTM are required to use consistent component specifications, parameter naming, and parameter units. However, support for any particular parameter of CTM is optional and software tools are free to support any subset of the CTM parameters that are suitable to their scope. Furthermore, software tools are free to develop their own data structures and data formats for encoding these parameters, as long as the parameter names and units are consistent. Extensions of the CTM specification on a application-by-application basis are welcomed and expected. Extensions are encouraged to follow the CTM style guidelines, when possible.

# 4 Conventions

CTM adheres to the following conventions, which also serve as guidelines for future extensions of CTM.

**Parameter Naming** Python's PEP8 Style Guide is leveraged as a foundational guide for the CTM specification. The most important features to highlight are that: (1) parameter names should be valid python identifiers (i.e. they cannot begin with numbers or special characters); (2) parameter names should be lower-case; (3) words in multi-word names are separated by underscores.

An additional parameter naming convention of CTM is that multi-word names should be ordered from most general to most specific. This provides a natural grouping of related parameters. For example, `cost_...` would defined a collection of parameters related to costs.

**Component Identifiers** Simple and unique identifiers are invaluable when processing data or when building multi-step analysis workflows. Integers are particularly well suited for such identifiers because they only require one computing word, leading to compact storage and fast comparisons. Consequently, every component in CTM is given a unique integer identifier, `uid`. These component index values must be unique within in each component class and they are not necessarily contiguous. The optional `name` field is also available for a more descriptive and human recognizable name of a component.

**Complex Numbers** Due to the oscillating nature of AC power systems, complex numbers provide a convenient mathematical tool for modeling these systems. Unfortunately, first-class complex numbers are not widely supported by modern programming languages, which precludes cross-language consistency. Due to this limitation, CTM adopts a real number encoding of complex network parameters. However, this raises the question of if those complex parameters should be encoded in rectangular or polar form. By default CTM adopts rectangular form as the standard for encoding complex numbers. However, there are a few cases where polar form is used (e.g. bus voltages and transformer taps) due to historical precedent and engineering practice.

To help support the encoding of complex numbers the following naming conventions are proposed by CTM specification. Let `C` be a complex number then the following short hands are used for the rectangular are polar forms of `C` respectively,

$$\mathtt{C} = \mathtt{cr} + \boldsymbol{i}\mathtt{ci} = \mathtt{cm}\angle\mathtt{ca}$$

The following shorthand is used for the conjugate product of a complex number `C`,

$$\mathtt{CC}^* = \mathtt{cr}^2 + \mathtt{ci}^2 = \mathtt{ccm}$$

**Parameters Types** The component parameters described in this model fall into one of three categories,

- *Static*, these are fixed values that describe the engineered proprieties and limitation of a device that are determined by the design and manufacturing of the component. This is adopted as the default parameter type as the majority of the parameter fall into this category.
- *Solution*, these are values that describe the state of a component at an instance in time. These values are usually the output of a power system analysis, however they can also be used as inputs in some cases; the most common case is when solving systems of equations, such as Power Flow.
- *Temporal Boundary Conditions*, temporal problems, such as Unit Commitment, require temporal boundary conditions describing the system's state before the current operating point. The post-fix `_prev` is used as a standard for specifying such boundary conditions.
- *Control System Settings*, these are values that describe the configuration of a local controller that companies a component. The post-fix `_setpoint` is used as a standard for specifying these parameters. A common example is the voltage setpoint of the governor inside of a generator.
- *Solution Initial Values*, in a variety of power system algorithms providing an initial solution or starting point can greatly increase performance. To that end, the post-fix `_start` can be added to *Solution* parameters to specify an inital point for an algorithm. A common example is setting the initial bus voltage profile for a Power Flow computation.

can be categorized into three characterize Some of the quantities in described in this model are indicated as *solution* parameters. These parameters are most often used to describe the output of

**Operating Ranges** A common feature of power system components are the permitted minimum and maximum values of a given control parameter. CTM adopts the nomenclature of lower bound (`lb`) and upper bound (`ub`) to encode these operating ranges. For example a complex number `C` can be bounded as follows,

$$\mathtt{cm\_lb} \leq \mathtt{cm} \leq \mathtt{cm\_ub}$$

$$\mathtt{ca\_lb} \leq \mathtt{ca} \leq \mathtt{ca\_ub}$$

$$\mathtt{cr\_lb} \leq \mathtt{cr} \leq \mathtt{cr\_ub}$$

$$\mathtt{ci\_lb} \leq \mathtt{ci} \leq \mathtt{ci\_ub}$$

In all cases lower bounds should always be less than or equal to upper bounds.

**Bus Connections and Orientation** Power networks are composed of a variety of components, with buses forming the coupling point where different components interact with each other. In the most general case, a network component can connect to an arbitrary number of buses in the network, as is the case with an $n$-winding transformer. However, all of the components considered by CTM connect to just one or two buses. In the case of a single bus connection, the parameter name `bus` is used to indicate the connecting bus. In the case of a two bus connection, the orientation can be critical (e.g. transformers). The terminology of from-bus (`bus_fr`) and to-bus (`bus_to`) is adopted to encode the orientation of two bus connections. In general, the shorthand `fr` and `to` are used to refer to quantities on the from and to sides of a component respectively.

**Boolean and Enumerated Values** Some parameters in power networks have a natural encoding as Boolean or enumerated data types. Notable examples are a component's `status` value, which indicates if it should be included or omitted in analysis, or the `bus_type` parameter, which can take one of three values PQ, PV, and Slack. A semantic encoding of these fields would be preferable from a data modeling standpoint. However, encoding all data model parameters as numeric values provides the advantage of a simple and consistent data type across multiple languages. This is particularly apparent when data is provided in a matrix form where all of the values must be numeric, as is the case in the Matpower [1] data format.

**Convention Exceptions for Power Systems** The guidelines above provide a consistent convention for naming new paramaters in power network models. However, a strong precedent already exists for some parameter names in power networks. The following list highlights key exceptions to convention in the interest of following long established conventions,

- $V$ - complex voltage
- $S = p + \boldsymbol{i}q$ - complex power
- $Z = r + \boldsymbol{i}x$ - complex impedance
- $Y = g + \boldsymbol{i}b$ - complex admittance

Consequently, the rectangular parameter names for complex power `S` that would be `sr`/`si` by the CTM conventions are instead replaced with their established names `p`/`q`. Similar replacements hold for impedance (`Z`) and admittance (`Y`) parameters.

# 5 The Common Electric Power Transmission System Model (CTM)

The CTM parameters fall into three categories: (1) global/network-wide parameters, which are consistent across all of the components in the network; (2) component capabilities parameters, which capture the basic operational limitations of a component and are independent of any specific operating point; and (3) solution parameters, which define a specific operating point of a network. Typically a data format will include one instantiation of each of these parameters. However, it is also reasonable that a data format would include multiple versions of the solution parameters to capture a variety of possible operating points.

In the interests of achieving the design goals outlined in this document the CTM format support the following foundational component types, buses, constant power loads, shunts, generators, storage, switches, AC lines, two-winding transformers, and two-terminal HVDC lines. Similar to the component parameters it is suitable for software adopting the CTM standards are welcome to extend the list of supported components to suit its needs.

The remainder of this section presents each of the CTM components and their associated parameters. The parameters of each component characterized by the following fields,

- name - The CTM standardized ASCII name for referencing this parameter.
- type - The abstract data type of the parameter.
- units - The standardized units of parameter.

- required - Indicates which target tasks (i.e. PF, OPF, and UC) require this parameter.
- description - A textual explanation of the parameter's function.

Additionally each component defines: additional requirements that the data must satisfy (beyond the given data type); mathematical symbols that are derived from the parameters; the canonical constraints that are defined for each component type.

## 5.1 Global Parameters

Table 1: Network Parameters

| name | type | units | required | description |
|---|---|---|---|---|
| `base_mva` | Float | MVA | always | scaling factor for per unit computations |
| `time_elapsed` | Float | hours | always | the amount of time that has passed since the previous time point for temporal boundary conditions. |
| `bus_ref` | Array of Int | none | OPF | a set of buses that set the reference angle (one per connected component) |

### Symbols

- $\Delta T =$ `time_elapsed`

## 5.2 Universal Component Parameters

Generic features

Table 2: Universal Component Parameters

| name | type | units | required | description |
|---|---|---|---|---|
| `uid` | Int | none | always | a unique identifier for components |
| `status` | Enum {0,1} | none | always | a 0/1 value indicating if the component should be included be omitted or not ($0 \Rightarrow$ omitted) |
| `name` | String | none | | a flexible name for components, non required to be unique |
| `source_uid` | unstructured | none | | encodes data for tracking the component ids from a source file (typically an Array of String) |

### Universal Symbols

- $i =$ `uid`

### Universal Data Requirements

- Any parameter that represents the magnitude of a complex number (e.g. `vm`, `sm`, `cm`, ...) should be positive.

## 5.3 Bus (`bus`)

### Symbols

- $V_i =$ `vm` $\angle$ `va`

Table 3: Bus Parameters

| name | type | units | required | description |
|---|---|---|---|---|
| base_kv | Float | kV | always | base voltage |
| type | Enum {1,2,3} | none | PF | bus type for power flow (1=PQ, 2=PV, 3=Slack) |
| vm_lb | Float | kV | OPF | a lower limit on voltage magnitude |
| vm_ub | Float | kV | OPF | an upper limit on voltage magnitude |
| area | Int | none | | assigned control area |
| zone | Int | none | | assigned control zone |
| Solution Data | | | | |
| vm | Float | kV | always | voltage magnitude |
| va | Float | degrees | always | voltage angle |

**Constraints**

- $\texttt{vm\_lb} \leq |V_i| \leq \texttt{vm\_ub}$
- TODO Power Balance

## 5.4 Load (load)

Table 4: Load Parameters

| name | type | units | required | description |
|---|---|---|---|---|
| bus | Int | none | always | connecting bus id |
| pd | Float | MW | always | active power demand |
| qd | Float | MVar | always | reactive power demand |
| pd_i | Float | MW | always | constant current active power demand |
| qd_i | Float | MVar | always | constant current reactive power demand |
| pd_y | Float | MW | always | constant admittance active power demand |
| qd_y | Float | MVar | always | constant admittance reactive power demand |

**Symbols**

- $b = \texttt{bus}$
- $S_i^{dp} = \texttt{pd} + \boldsymbol{i}\texttt{qd}$
- $S_i^{di} = (\texttt{pd\_i} + \boldsymbol{i}\texttt{qd\_i})|V_b|$
- $S_i^{dy} = (\texttt{pd\_y} + \boldsymbol{i}\texttt{qd\_y})^*|V_b|^2$

**Constraints**

- $S_i^d = S_i^{dp} + S_i^{di}|V_b| + (S_i^{dy})^*|V_b|^2$

## 5.5 Shunt (shunt)

**Symbols**

- $Y_i^s = \texttt{gs} + \boldsymbol{i}\texttt{bs}$

Table 5: Shunt Parameters

| name | type | units | required | description |
|---|---|---|---|---|
| `bus` | Int | none | always | connecting bus id |
| `gs` | Float | MW at 1.0 V p.u. Ohm? | always | active power demand |
| `bs` | Float | MVar at 1.0 V p.u. Ohm? | always | reactive power demand |

### Constraints

- none?

## 5.6 Generator (gen)

### Symbols

- $S_i^g = \texttt{pg} + \boldsymbol{i}\texttt{qg}$

### Constraints

- $\texttt{pg\_lb} + \boldsymbol{i}\texttt{qg\_lb} \leq S^g \leq \texttt{pg\_ub} + \boldsymbol{i}\texttt{qg\_ub}$
- TODO cost model
- TODO commitment constraints
- TODO cost_start by time (startup hot (9h), warm (9-16h), cold (16h))

## 5.7 Storage (storage)

Detailed derivation and motivation for this model can be found in [].

### Data Requirements

- `charge_efficiency` is strictly positive
- `discharge_efficiency` is strictly positive

### Symbols

- $b = \texttt{bus}$
- $S_i^s = \texttt{ps} + \boldsymbol{i}\texttt{qs}$
- $S_i^s = V_b I_i^s$
- $S_i^{ex} = \texttt{ps\_ex} + \boldsymbol{i}\texttt{qs\_ex}$
- $S_i^e = \texttt{pe} + \boldsymbol{i}\texttt{qe}$
- $\eta_i^c = \texttt{charge\_efficiency}$
- $\eta_i^d = \texttt{discharge\_efficiency}$
- $pc_i$ - charge value
- $pd_i$ - discharge value

Table 6: Generator Parameters

| name | type | units | required | description |
|---|---|---|---|---|
| `bus` | Int | none | always | connecting bus id |
| `vm_setpoint` | Float | kV | PF | the target voltage magnitude of the bus that this generator is connected to |
| `pg_lb` | Float | MW | OPF, UC | minimum active power generation |
| `pg_ub` | Float | MW | OPF, UC | maximum active power generation |
| `qg_lb` | Float | MVar | OPF, UC | minimum reactive power generation |
| `qg_ub` | Float | MVar | OPF, UC | maximum reactive power generation |
| `cost_pg_model` | Enum {1,2} | none | OPF, UC | cost model type (1=pwl, 2=polynomial) |
| `cost_pg_parameters` | Array of Float | $/MWh | OPF, UC | either points for a pwl cost (mw,cost) or polynomial coefficients (from lowest degree first) |
| `startup_cost_hot` | Float | $ | UC | cost in $ of starting a unit |
| `startup_cost_warm` | Float | $ | UC | cost in $ of starting a unit |
| `startup_cost_cold` | Float | $ | UC | cost in $ of starting a unit |
| `startup_time_hot` | Float | hours | UC | threshold of time (est. 9h) |
| `startup_time_warm` | Float | hours | UC | threshold of time (est. 9-16h) |
| `startup_time_cold` | Float | hours | UC | threshold of time (est. 16h+) |
| `in_service_time_ub` | Float | hours | UC | the maximum amount of time a unit can be in service |
| `in_service_time_lb` | Float | hours | UC | the minimum amount of time a unit can be in service |
| `down_time_lb` | Float | hours | UC | the minimum amount of time a unit must be out of service |
| `pg_delta_ub` | Float | MW/h | UC | maximum active power increase per hour |
| `pg_delta_lb` | Float | MW/h | UC | maximum active power decrease per hour |
| `service_required` | Enum {0,1,2} | none | UC | fixes the service value of the unit (0 = no requirement, 1 = fixed in service, 2 = fixed out of service) |
| Temporal Boundary Data | | | | |
| `pg_prev` | float | MW | UC | previous active power output |
| `in_service_time_prev` | float | hours | UC | time up previously |
| `down_time_prev` | float | hours | UC | time up previously |
| Solution Data | | | | |
| `pg` | Float | MW | always | active/real power generation output |
| `qg` | Float | MVar | always | reactive/imaginary power generation output |
| `in_service` | Bool | none | UC | true if the unit is in service |

**Constraints**

- $S_i^s + S_i^e = S_i^{ex}$
- $\Re(S_i^e) - \texttt{energy\_prev} = \Delta T \left( \eta_i^c pc_i - \frac{pd_i}{\eta_i^d} \right)$
- $pc_i \cdot pd_i = 0.0$
- $0 \leq e_i \leq \texttt{energy\_ub}$

Table 7: Storage Parameters

| name | type | units | required | description |
|---|---|---|---|---|
| `bus` | Int | none | always | connecting bus id |
| `charge_efficiency` | Float | none | always | captures losses due to charging (non-zero) |
| `discharge_efficiency` | Float | none | always | captures losses due to discharging (non-zero) |
| `ps_ex` | Float | MW | always | converter standby active power exogenous flow |
| `qs_ex` | Float | MVar | always | converter standby reactive power exogenous flow |
| `energy_ub` | Float | MWh | OPF | maximum state of charge |
| `charge_ub` | Float | MW | OPF | maximum charge rating |
| `discharge_ub` | Float | MW | OPF | maximum discharge rating |
| `sm_ub` | Float | MVA | OPF | converter apparent power rating |
| `cm_ub` | Float | MA | OPF | converter current output rating |
| `qs_lb` | Float | MVar | OPF | minimum reactive power injection |
| `qs_ub` | Float | MVar | OPF | maximum reactive power injection |
| `ps_delta_ub` | Float | MW/h | UC | maximum active power increase per hour |
| `ps_delta_lb` | Float | MW/h | UC | maximum active power decrease per hour |
| Temporal Boundary Data | | | | |
| `energy_prev` | Float | MWh | always | initial state of charge |
| Solution Data | | | | |
| `ps` | Float | MW | always | active/real power storage injection |
| `qs` | Float | MVar | always | reactive/imaginary power storage injection |
| `energy` | Float | MWh | always | current state of charge |
| `charge` | Float | MW | | charging |
| `discharge` | Float | MW | | discharging |

- $0 \leq pc_i \leq$ `charge_ub`
- $0 \leq pd_i \leq$ `discharge_ub`
- $|S_i^s| \leq$ `sm_ub`
- $|I_i^s| \leq$ `cm_ub`
- `qs_lb` $\leq \Im(S_i^s) \leq$ `qs_ub`

## 5.8 Switch (`switch`)

**Symbols**

- $f =$ `bus_fr`
- $t =$ `bus_to`
- $S_i^w =$ `psw` $+ \boldsymbol{i}$`qsw`
- $S_i^w = V_f(I_i^w)^* = V_t(I_i^w)^*$

Table 8: Switch Parameters

| name | type | units | required | description |
|---|---|---|---|---|
| `bus_fr` | Int | none | always | from-side connecting bus |
| `bus_to` | Int | none | always | to-side connecting bus |
| `sm_ub` | Float | MVA | OPF | apparent power flow limit |
| `cm_ub` | Float | MA | OPF | current flow limit |
| Solution Data | | | | |
| `psw_fr` | Float | MW | always | active/real power flow |
| `qsw_fr` | Float | MVar | always | reactive/imaginary power flow |
| `state` | Enum {0,1} | none | always | open/closed state, 0 ⇒ open, 1 ⇒ closed |

**Constraints**

- $|S_i^w| \leq$ `sm_ub`
- $|I_i^w| \leq$ `cm_ub`
- `state` $= 0 \Rightarrow |S_i^w| = 0 \wedge |I_i^w| = 0$
- `state` $= 1 \Rightarrow V_f = V_t$

### 5.9 AC Line (`ac_line`)

Table 9: AC Line Parameters

| name | type | units | required | description |
|---|---|---|---|---|
| `bus_fr` | Int | none | always | from-side connecting bus |
| `bus_to` | Int | none | always | to-side connecting bus |
| `r` | Float | Ohm | always | line resistance |
| `x` | Float | Ohm | always | line reactance |
| `g_fr` | Float | Ohm | always | line charge conductance, from side |
| `b_fr` | Float | Ohm | always | line charge susceptance, from side |
| `g_to` | Float | Ohm | always | line charge conductance, to side |
| `b_to` | Float | Ohm | always | line charge susceptance, to side |
| `sm_ub_a` | Float | MVA | OPF | persistent apparent power limit |
| `sm_ub_b` | Float | MVA | OPF | 4 hour apparent power limit |
| `sm_ub_c` | Float | MVA | OPF | 15 minute emergency apparent power limit |
| `cm_ub_a` | Float | MA | OPF | persistent current limit |
| `cm_ub_b` | Float | MA | OPF | 4 hour current limit |
| `cm_ub_c` | Float | MA | OPF | 15 minute emergency current limit |
| `vad_lb` | Float | degrees | OPF | voltage angle difference limit |
| `vad_ub` | Float | degrees | OPF | voltage angle difference limit |
| Solution Data | | | | |
| `pl_fr` | Float | MW | | active power flow, from side |
| `ql_fr` | Float | MVar | | reactive power flow, from side |
| `pl_to` | Float | MW | | active power flow, to side |
| `ql_to` | Float | MVar | | reactive power flow, to side |

**Data Requirements**

- The `base_kv` values of `bus_fr` and `bus_to` should be the same

**Symbols**

- $f =$ bus_fr
- $t =$ bus_to
- $Z_i =$ r $+ \boldsymbol{i}$x
- $Y_i = 1/Z_i$
- $Y_i^{cf} =$ g_fr $+ \boldsymbol{i}$b_fr
- $Y_i^{ct} =$ g_to $+ \boldsymbol{i}$b_to
- $S_i^{lf} =$ pl_fr $+ \boldsymbol{i}$ql_fr
- $S_i^{lt} =$ pl_to $+ \boldsymbol{i}$ql_to

**Constraints**

- $S_i^{lf} = \left(Y_i + Y_i^{cf}\right)^* |V_f|^2 - Y_i^* V_f V_t^*$
- $S_i^{lt} = \left(Y_i + Y_i^{ct}\right)^* |V_t|^2 - Y_i^* V_t V_f^*$
- $|S_i^{lf}| \leq$ sm_ub_*
- $|S_i^{lt}| \leq$ sm_ub_*
- $|I_i^{lf}| \leq$ cm_ub_*
- $|I_i^{lt}| \leq$ cm_ub_*
- vad_lb $\leq \angle(V_f (V_t)^*) \leq$ vad_ub

### 5.10 Transformer (transformer)

Two-Winding T-Model Transformer. Multiple wingdings can be implemented by adding internal *star* bus. do emergency ratings apply in this case as well? Can vad bounds be justified?

**Symbols**

- $f =$ bus_fr
- $t =$ bus_to
- $S_i^{tf} =$ pt_fr $+ \boldsymbol{i}$qt_fr
- $S_i^{tt} =$ pt_to $+ \boldsymbol{i}$qt_to
- $Z_i =$ r $+ \boldsymbol{i}$x
- $Y_i = 1/Z_i$
- $T_i =$ tm $\angle$ ta
- $Y_i^c =$ g $+ \boldsymbol{i}$b
- $tm^s =$
- $ta^s =$

Table 10: Transformer Parameters

| name | type | units | required | description |
|---|---|---|---|---|
| `bus_fr` | Int | none | always | from-side connecting bus |
| `bus_to` | Int | none | always | to-side connecting bus |
| `r` | Float | Ohm | always | line resistance |
| `x` | Float | Ohm | always | line reactance |
| `g` | Float | Ohm | always | internal charge conductance |
| `b` | Float | Ohm | always | internal charge susceptance |
| `tm_lb` | Float | Ohm | OPF | minimum tap ratio value |
| `tm_ub` | Float | Ohm | OPF | maximum tap ratio value |
| `tm_steps` | Int | none | OPF | the number of discrete steps between `tm_lb` and `tm_ub` |
| `ta_lb` | Float | Ohm | OPF | minimum phase angle shift value |
| `ta_ub` | Float | Ohm | OPF | maximum phase angle shift value |
| `ta_steps` | Int | none | OPF | the number of discrete steps between `ta_lb` and `ta_ub` |
| `sm_ub_a` | Float | MVA | OPF | persistent apparent power limit |
| `sm_ub_b` | Float | MVA | OPF | 4 hour apparent power limit |
| `sm_ub_c` | Float | MVA | OPF | 15 minute emergency apparent power limit |
| `cm_ub_a` | Float | MA | OPF | persistent current limit |
| `cm_ub_b` | Float | MA | OPF | 4 hour current limit |
| `cm_ub_c` | Float | MA | OPF | 15 minute emergency current limit |
| Solution Data | | | | |
| `tm` | Float | Ohm | always | tap ratio |
| `ta` | Float | Ohm | always | phase angle shift |
| `pt_fr` | Float | MW | | active power flow, from side |
| `qt_fr` | Float | MVar | | reactive power flow, from side |
| `pt_to` | Float | MW | | active power flow, to side |
| `qt_to` | Float | MVar | | reactive power flow, to side |

### Constraints

- TODO double check these equations
- $S^{tf}_i = Y^{c*}_i |V_f|^2 + Y^*_i \frac{|V_f|^2}{|T_i|^2} - Y^*_i \frac{V_f V_t^*}{T_i}$
- $S^{tt}_i = Y^*_i |V_t|^2 - Y^*_i \frac{V_t V_f^*}{T_i^*}$
- $|S^{tf}_i| \leq \texttt{sm\_ub\_*}$
- $|S^{tt}_i| \leq \texttt{sm\_ub\_*}$
- $|I^{tf}_i| \leq \texttt{cm\_ub\_*}$
- $|I^{tt}_i| \leq \texttt{cm\_ub\_*}$
- $\texttt{tm\_lb} \leq |T_i| \leq \texttt{tm\_ub}$
- $(|T_i| \cdot \texttt{tm\_steps})/(\texttt{tm\_ub} - \texttt{tm\_lb}) \in \mathbb{Z}$
- $\texttt{ta\_lb} \leq \angle(T_i) \leq \texttt{ta\_ub}$
- $(\angle(T_i) \cdot \texttt{ta\_steps})/(\texttt{ta\_ub} - \texttt{ta\_lb}) \in \mathbb{Z}$
- $\texttt{vad\_lb} \leq \angle(V_f (V_t)^*) \leq \texttt{vad\_ub}$

## 5.11 Point-to-point HVDC (`hvdc_p2p`)

This section defines parameters for point-to-point HVDC connections with simplified converter stations. Detailed derivation and motivation for the underlying model can be found in [4]. The main simplifications are:

- converter technology and ratings identical at from and to side
- loss factors are divided by two before assigning to each converter
- the current limit of the dc line is enforced through the current limit of the converters
- no filters or transformers at the converter stations

Further simplifications are possible:

- dropping lossb and lossc
- dropping dc line resistance
- dropping dc-side voltage +bounds

TODO: Check out data model of [5][6] and contact them?

### Symbols

- $f =$ `bus_fr`
- $t =$ `bus_to`
- $p_i =$ `p`
- $R_i =$ `r`
- $G_i = 1/R_i$
- $S_i^{df} = P_i^{df} + \boldsymbol{i}Q_i^{df} = \texttt{pdc\_fr} + \boldsymbol{i}\texttt{qdc\_fr}$
- $S_i^{dt} = P_i^{dt} + \boldsymbol{i}Q_i^{dt} = \texttt{pdc\_to} + \boldsymbol{i}\texttt{qdc\_to}$
- $V_i^{dc,f}$ dc voltage at from side converter
- $V_i^{dc,t}$ dc voltage at to side converter
- $a_i =$ loss_a
- $b_i =$ loss_b
- $c_i =$ loss_c
- $\phi_i^f$ firing angle of the from side LCC converter
- $\phi_i^t$ firing angle of the to side LCC converter
- $P_i^{dc,f}$ dc-side power
- $Q_i^{dc,f}$ reactive power slack variable in converter

[6] https://hynet.readthedocs.io/en/latest/usage.html#management-of-grid-databases

Table 11: HVDC Line Parameters

| name | type | units | required | description |
|---|---|---|---|---|
| `bus_fr` | Int | none | always | from-side connecting bus |
| `bus_to` | Int | none | always | to-side connecting bus |
| `base_kv_dc` | Float | kV | always | base voltage at the dc side |
| `vm_dc_lb` | Float | kV | OPF | a lower limit on dc voltage magnitude |
| `vm_dc_ub` | Float | kV | OPF | an upper limit on dc voltage magnitude |
| `pdc_fr_lb` | Float | MW | OPF | minimum active power flow, from side |
| `qdc_fr_lb` | Float | MVar | OPF | minimum reactive power flow, from side |
| `pdc_fr_ub` | Float | MW | OPF | maximum active power flow, from side |
| `qdc_fr_ub` | Float | MVar | OPF | maximum reactive power flow, from side |
| `pdc_to_lb` | Float | MW | OPF | minimum active power flow, to side |
| `qdc_to_lb` | Float | MVar | OPF | minimum reactive power flow, to side |
| `pdc_to_ub` | Float | MW | OPF | maximum active power flow, to side |
| `qdc_to_ub` | Float | MVar | OPF | maximum reactive power flow, to side |
| `r` | Float | Ohm | always | dc line resistance |
| `p` | Enum {1, 2} | none | always | number of poles (1=monopole, 2=bipole) |
| `technology` | Enum {1, 2, 3} | none | always | power conversion technology (1=LCC, 2=VSC, 3=MMC) |
| `loss_a` | Float | MVA | always | standby loss |
| `loss_b` | Float | kV | always | loss proportional to current magnitude |
| `loss_c` | Float | Ohm | always | loss proportional to current magnitude squared |
| `sm_ub` | Float | MVA | OPF | apparent power limit |
| `cm_ub` | Float | MA | OPF | current limit |
| `phi_lb` | Float | degrees | OPF | if LCC: firing angle minimum |
| `phi_ub` | Float | degrees | OPF | if LCC: firing angle maximum |
| Solution Data | | | | |
| `vm_dc` | Float | kV | always | voltage magnitude at the dc side |
| `pdc_fr` | Float | MW | | active power flow, from side |
| `qdc_fr` | Float | MVar | | reactive power flow, from side |
| `pdc_to` | Float | MW | | active power flow, to side |
| `qdc_to` | Float | MVar | | reactive power flow, to side |

**Constraints**

- $P_i^{dc,f} = G_i p_i V_i^{dc,f}(V_i^{dc,f} - V_i^{dc,t})$
- $P_i^{dc,t} = G_i p_i V_i^{dc,t}(V_i^{dc,t} - V_i^{dc,f})$
- $S_i^{df} - P_i^{dc,f} + \boldsymbol{i}Q_i^{dc,f} = a_i/2 + b_i/2|I_i^{df}| + c_i/2|I_i^{df}|^2$
- $S_i^{dt} - P_i^{dc,t} + \boldsymbol{i}Q_i^{dc,t} = a_i/2 + b_i/2|I_i^{dt}| + c_i/2|I_i^{dt}|^2$
- $|S_i^{df}| \leq \texttt{sm\_ub}$
- $|S_i^{dt}| \leq \texttt{sm\_ub}$
- $-\texttt{sm\_ub} \leq Q_i^{dc,f} \leq \texttt{sm\_ub}$
- $-\texttt{sm\_ub} \leq Q_i^{dc,t} \leq \texttt{sm\_ub}$
- $\texttt{pdc\_fr\_lb} \leq P_i^{df} \leq \texttt{pdc\_fr\_ub}$

- $\texttt{pdc\_to\_lb} \leq P_i^{dt} \leq \texttt{pdc\_to\_ub}$
- $\texttt{qdc\_fr\_lb} \leq Q_i^{df} \leq \texttt{qdc\_fr\_ub}$
- $\texttt{qdc\_to\_lb} \leq Q_i^{dt} \leq \texttt{qdc\_to\_ub}$
- $|I_i^{df}| \leq \texttt{cm\_ub}$
- $|I_i^{dt}| \leq \texttt{cm\_ub}$
- $\texttt{vm\_dc\_lb} \leq V_i^{dc,f} \leq \texttt{vm\_dc\_ub}$
- $\texttt{vm\_dc\_lb} \leq V_i^{dc,t} \leq \texttt{vm\_dc\_ub}$

if LCC:

- $Q_i^{df} \geq 0$
- $Q_i^{dt} \geq 0$
- $0 \leq \texttt{phi\_lb} \leq \phi_i^f \leq \texttt{phi\_ub} \leq \pi$
- $0 \leq \texttt{phi\_lb} \leq \phi_i^t \leq \texttt{phi\_ub} \leq \pi$
- $P_i^{dc,f} = \cos(\phi_i^f)\texttt{sm\_ub}$
- $Q_i^{dc,f} = \sin(\phi_i^f)\texttt{sm\_ub}$
- $P_i^{dc,t} = \cos(\phi_i^t)\texttt{sm\_ub}$
- $Q_i^{dc,t} = \sin(\phi_i^t)\texttt{sm\_ub}$

## 5.12 Reserve (`reserve`)

Three pre-defined forms of reserve are allowed,

- *AGC*, Covers support from 0 to 10 minutes
- *Spin*, Covers support from 10 to 20 minutes
- *Flex*, Covers support over 20 minutes

Table 12: Reserve Parameters

| name | type | units | required | description |
|---|---|---|---|---|
| `reserve_type` | ENUM {1,2,3} | none | always | 1=AGC, 2=Spin, 3=Flex |
| `participants` | Array of Int | none | UC | A list of generator uids that will respond to this reserve. |
| `pg_lb` | Float | MW | UC | minimum active power required by this reserve. |
| `pg_ub` | Float | MW | UC | maximum active power required by this reserve. |

**Symbols**

- TODO

**Constraints**

- TODO

# 6 TODOs

- ZIP loads, should shunt parameters values given as Z or Y? Verify mathmatical model.
- Consensus of terminology of ”network element” or ”network component”; maybe standardize about element (lead, Zimmerman)
- add remark about avoiding name clashes in the global scope, motivates (`pg`,`qg` over `p`,`q`)
- Add note data correctness checks that preclude basic mathematical issues (e.g. voltage magnitudes should be positive) (Coffrin, Geth)
- Should generators have an internal voltage source and internal impedance? Seems inconsistent that storage and HVDC has this, but not generators. One suggestion is ignore internal losses in this document version and delay this for a future iteration.
- add fields for ZIP loads
- Zonal Reserves, need to add (lead, Coffrin)
- generic model for reserve products (JP / Ben)
- Add some semantics for bus areas and zones (Coffrin, Geth)
- Add recommended Per-Unit conversion to the document (lead, Geth)
- Add proper unit-commitment data (Coffrin, Ben)?
- min/max may be preferable to lb/ub (Fobes/JP)
- resolve generator type / fuel type data
- Feedback from PandaPower
- Reach out to PyPSA folks (Clayton can help), ANL UC.jl package; HELICS (trevor.hardy@pnnl.gov)
- Reach out to Yung Hong?

## 6.1 Discussion Points

The following items were raised for discussion and consideration by the broader team.

- should some kine of zones or other collection be used to define base kv values? To reduce the data errors?
- Idealized transformer components vs lossy transformers (Claeys, Geth)
- Would an explicit 3-winding (or n-winding) transformer model provide additional value? (Claeys, Fobes)
- What is the proposed method for zero impedance transformers (and lines?). (Claeys)
- feature prioritization

# A Feature Roadmap

## A.1 v1.0.0

- Initial release

## A.2 v1.1.0

- Add fields for indicating controllable vs fixed components (lead, Coffrin; requested by many folks)
- can we provide standard for multiple time points via change data; support for arbitrary delta (Matpower Change tables; replacement, scaling, additive; individual row, all rows, zone) (lead, Coffrin; requested by many folks)
- increase model fidelity by adding internal voltage source and internal impedance to generators; and similar modeling fidelity to storage and HVDC lines.
- shunt properties, fixed, continuous, discrete, cap banks, ect...
- price responsive demands (lead Zimmerman)
- non-spinning reserves (a 30-minute product), (lead Ben)
- add fule_type and prime_mover feilds to generators, (lead Clayton)
- contingency lists

**Discussion Points**

- TBD

## A.3 v1.2.0

- Should ZIP/exponential load models (Claeys)

**Discussion Points**

- TBD

## A.4 Version TBD

- Add fields for indicating controllable vs fixed components (lead, Zimmerman/Coffrin)
- change storage name from charge/discharge to input/output
- Document that omitted values and Inf / -Inf can be used to define inactive bounds
-
- Explicit model of renewable gens (lead, Ben/JP)
- more detailed shunt models (cap banks, UPFCs, synchronous condensors, SVCs, STATCOMs)
- controlable loads
- Generator D-Curves
- HVDC Buses
- time dependent line flow limits
- tap dependent transformer parameters
- indication of important / unimportant line flow constraints
- remote monitor points for control systems (Ray)

**Discussion Points**

- TBD

# References


[1] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, “MATPOWER: steady-state operations, systems research and education,” *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 12–19, 2011.

[2] S. Meinecke, M. Braun, F. Meier, A. Scheidler, F. Schafer, J. Dollichon, L. Thurner, and J.-H. Menke, “Pandapower - An open-source python tool for convenient modeling, analysis, and optimization of electric power systems,” *IEEE Trans. Power Syst.*, vol. 33, no. 6, pp. 6510–6521, 2018.

[3] C. Coffrin, R. Bent, K. Sundar, Y. Ng, and M. Lubin, “PowerModels.jl: an open-source framework for exploring power flow formulations,” in *Power Syst. Comp. Conf.*, vol. 20, Dublin, Ireland, 2018, p. 8.

[4] H. Ergun, J. Dave, D. Van Hertem, and F. Geth, “Optimal power flow for AC/DC Grids: formulation, convex relaxation, linear approximation and implementation,” *IEEE Trans. Power Syst.*, vol. 34, no. 4, pp. 2980 – 2990, 2019.

[5] M. Hotz and W. Utschick, “hynet: an optimal power flow framework for hybrid AC/DC power systems,” *IEEE Trans. Power Syst.*, vol. XX, no. X, pp. 1–12, 2019. [Online]. Available: http://arxiv.org/abs/1811.10496