CHANGING WHAT'S POSSIBLE

High Fidelity, Year Long Power Network Data Sets for Replicable Power System Research

Pascal Van Hentenryck
University of Michigan

Overview

- The team
- Project objectives
- Technical approach
- Accomplishments
- Goals
- Tech-to-Market
- Collaborations

Team

- Dan Bienstock, Columbia
  - IOE, optimization, discrete mathematics
- James Anderson, Steven Low, Caltech
  - EECS, control, optimization
- Carleton Coffrin, Russell Bent, LANL
  - CS, optimization
- Ian Hiskens, Ferdinando Fioretto, Pascal Van Hentenryck
  - CS, EECS, IOE
- Patrick Panciatici, RTE
  - all of the above

Overview

- The team
- ***Project objectives***
- Technical approach
- Accomplishments
- Goals
- Tech-to-Market
- Collaborations

Overview (1)

- High-fidelity modeling of electricity networks
  - components + control systems
- JSON format to capture the required fidelity
  - useful for operators and optimization software
- Real test cases coming from RTE
  - largest transmission operator in Europe
- Transformation tools and scripting
  - node breaker, bus breaker, bus branch
  - per unit transformations
  - rollback

Overview (2)

- Validation, quality control, and metrics
  - validation of modeling and parameters
  - characterization of problem complexity
- Synthetic test cases
  - graph manipulation, exploiting the structure
- Disaggregation and obfuscation
  - preserving privacy in time series
- Repository
  - replicable research

Overview

- The team
- Project objectives
- ***Technical approach***
- Accomplishments
- Goals
- Tech-to-Market
- Collaborations

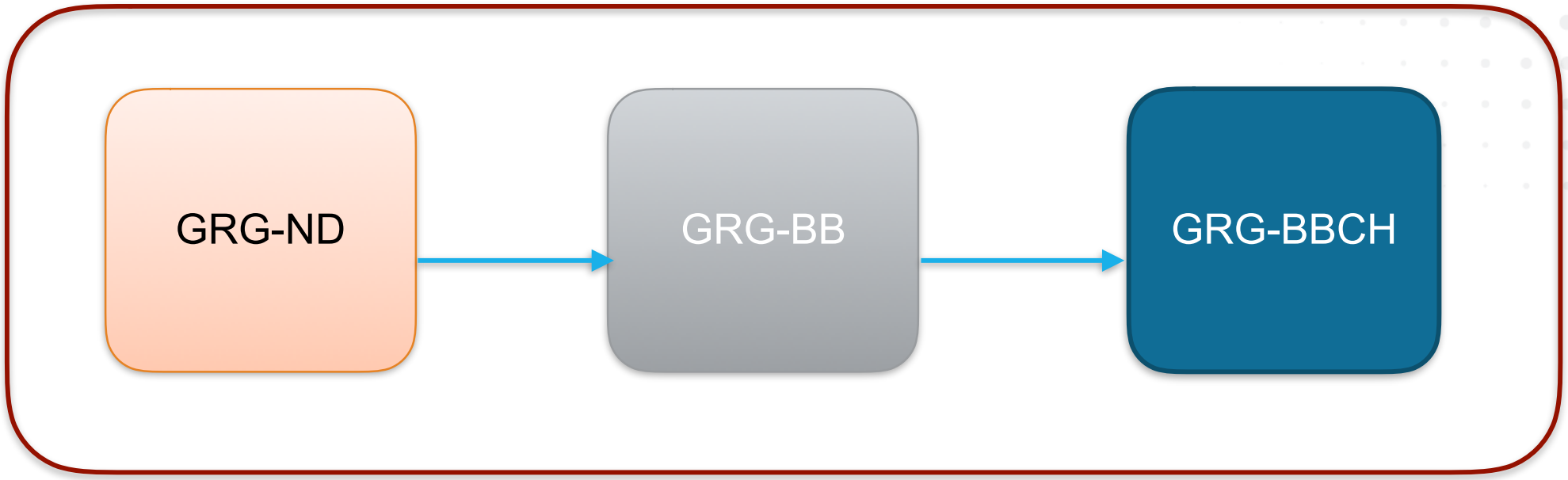
JSON Format

- Capture electrical power system with high fidelity
  - components, constraints, corrective actions, …
- Cognizant of computational issues
  - the distance to solvers must be small
  - the translation should be efficient computationally
- Extensibility
  - the format can easily accommodate new components, attributes, …
- Standard, modern, and reusable technology
  - JSON is becoming ubiquitous

Targeted Audience

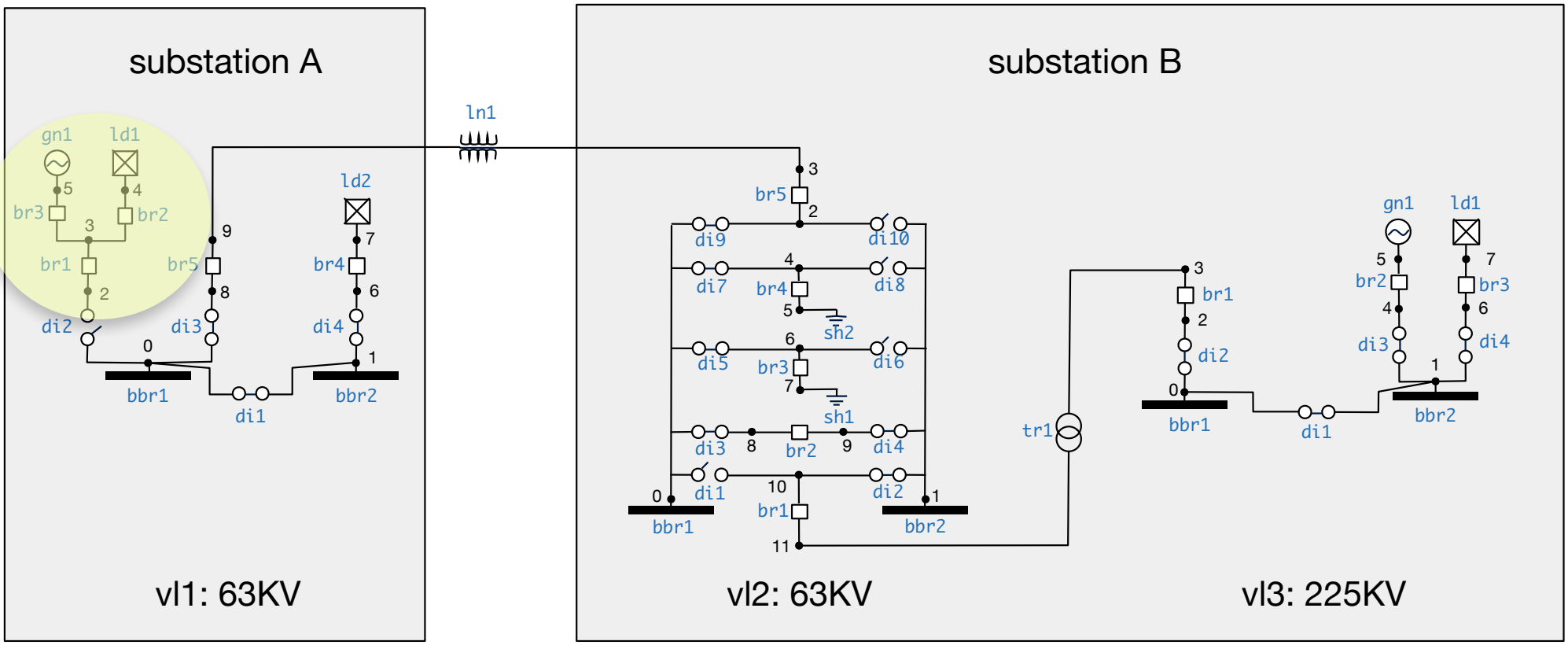
- No unique audience
  - operators running the system
  - developers of the underlying solvers
- Operators
  - would like to see the physical organization of the network
    - A substation is a meaningful concept and should be preserved
  - would like to see actual units for
    - e.g., voltage magnitudes
- Tool developers
  - would like to start from a representation which is as close as possible to the solver needs

Philosophy

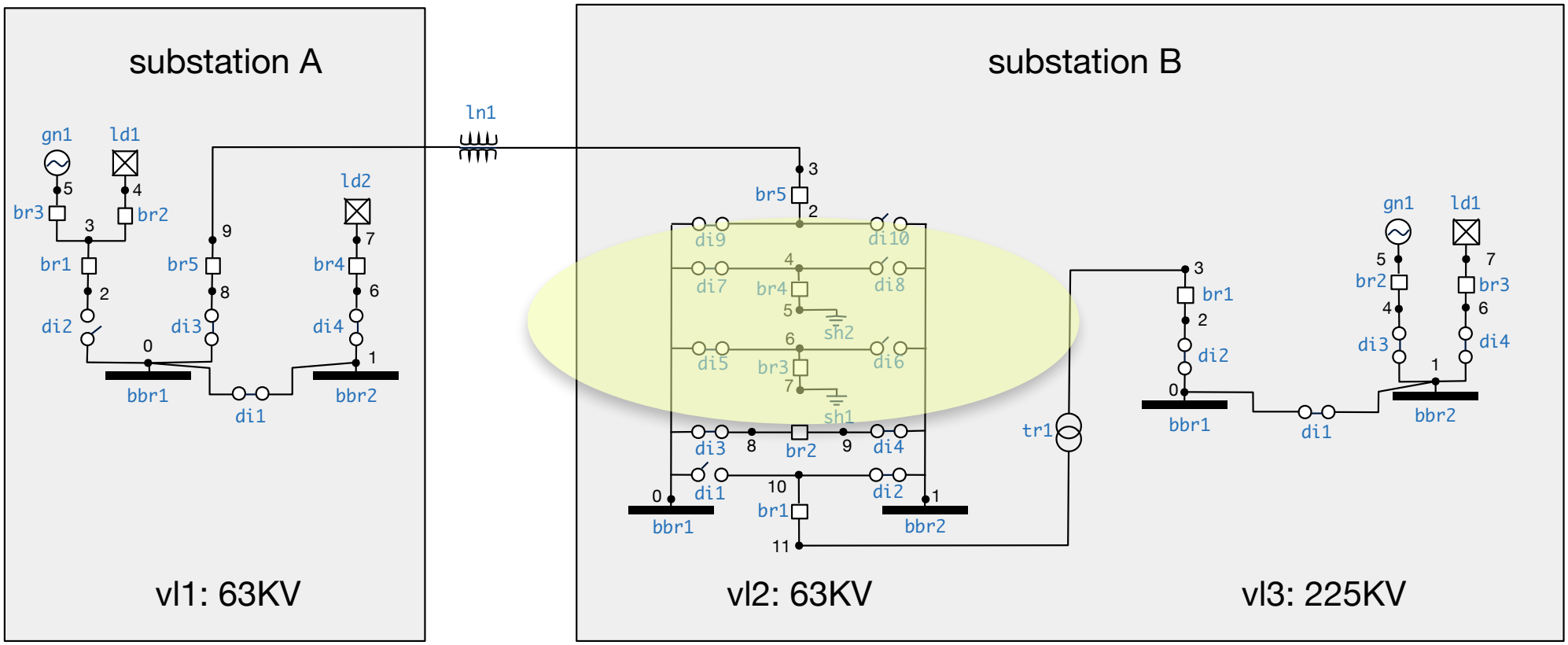


Node Breaker Bus Breaker Bus Branch

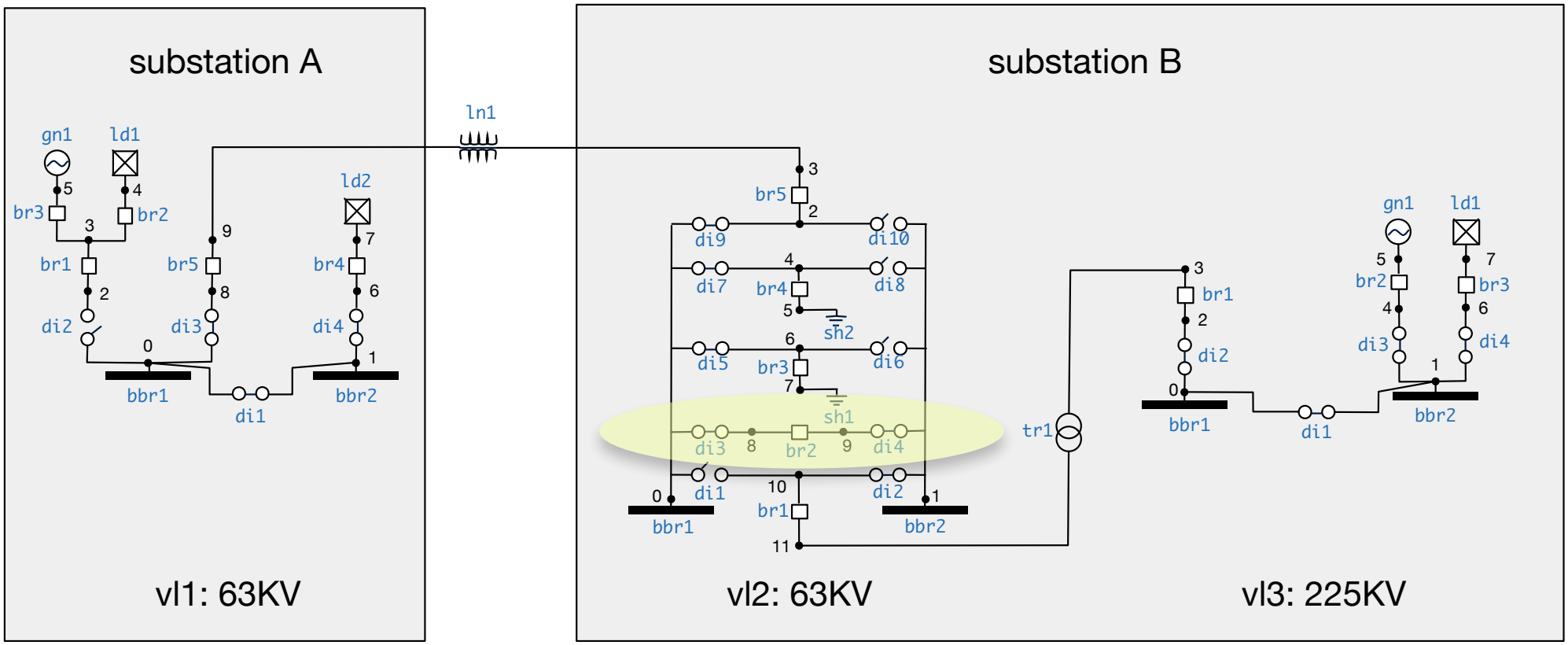
GRG-NB: The Physical Network



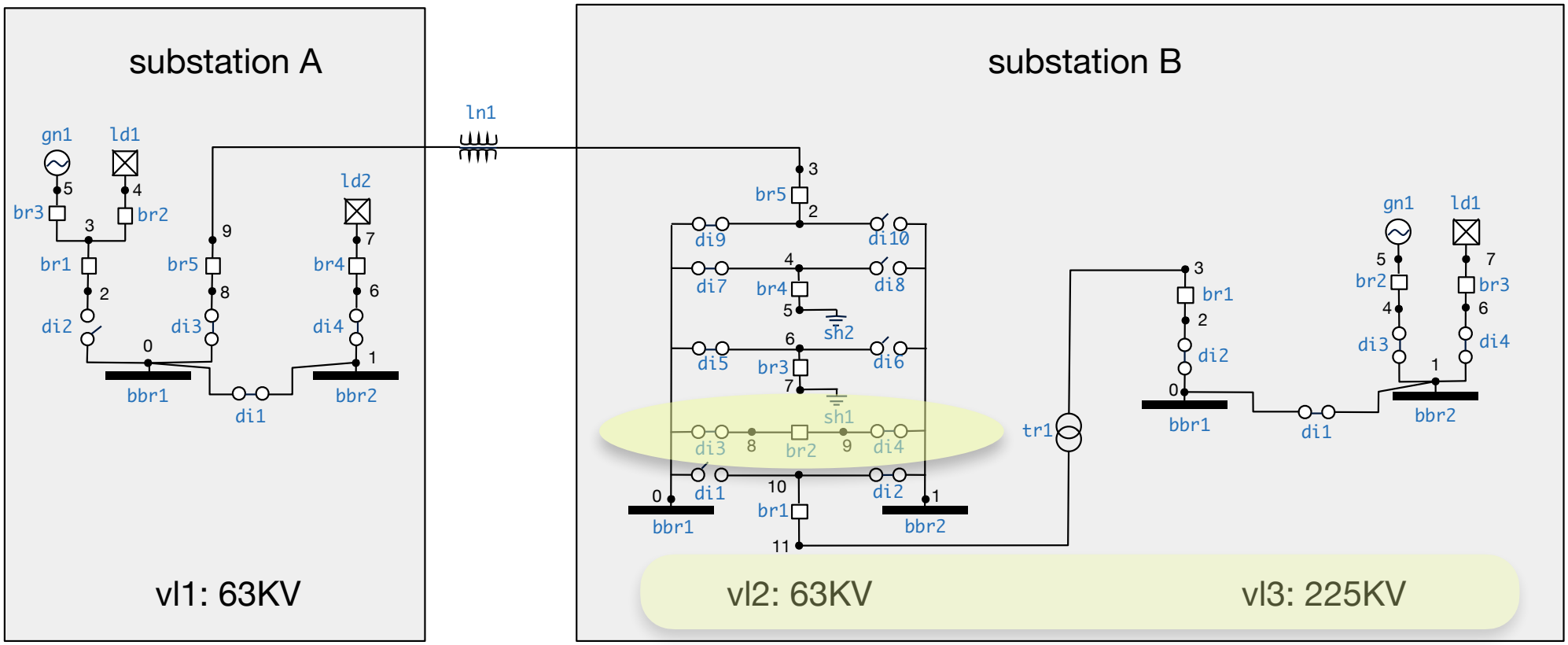
GRG-NB: The Physical Network



GRG-NB: The Physical Network



GRG-NB: The Physical Network



GRG Formats

	Node-breaker	Bus-breaker	Bus-branch
Topology	yes	yes	no
Breakers	yes	yes	no
Disconnectors	yes	no	no
Bus type	busbar	logical bus	bus

GRG Formats

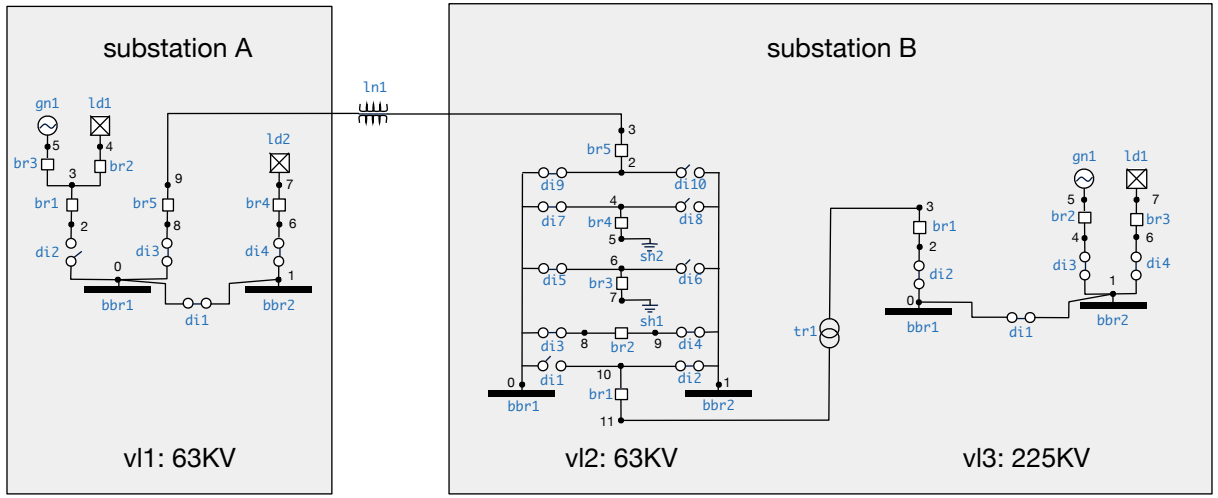
	Node-breaker	Bus-breaker	Bus-branch
Topology	yes	yes	no
Breakers	yes	yes	no
Disconnectors	yes	no	no
Bus type	busbar	logical bus	bus

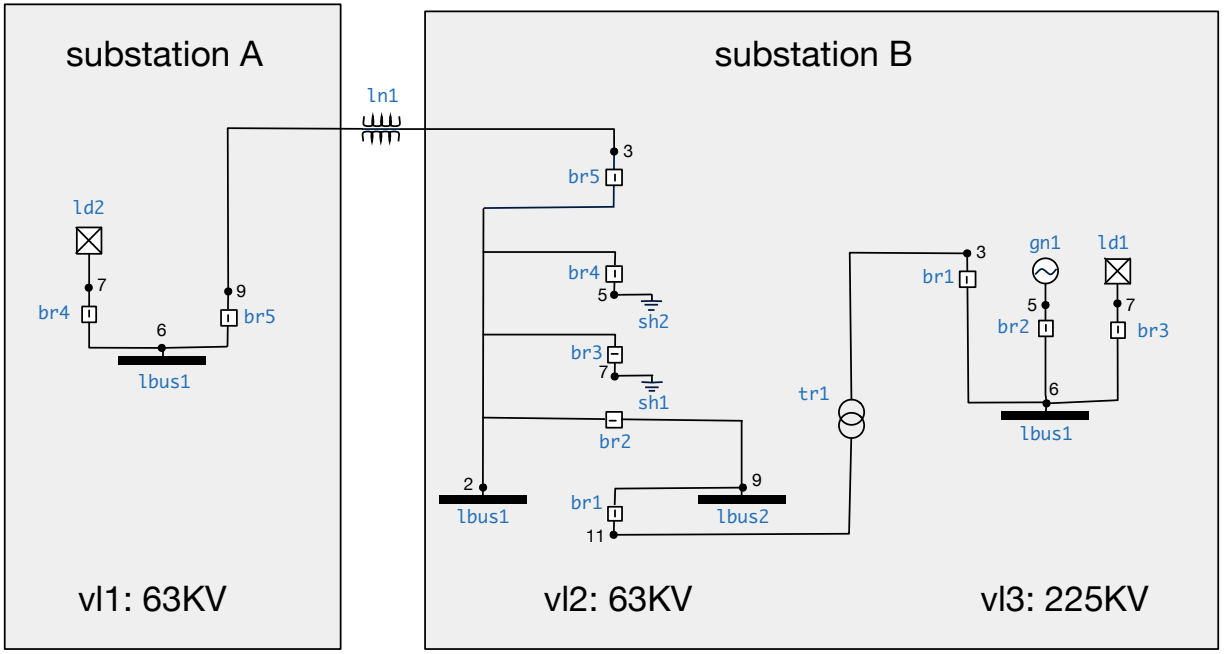
GRG Formats

	Node-breaker	Bus-breaker	Bus-branch
Topology	yes	yes	no
Breakers	yes	yes	no
Disconnectors	yes	no	no
Bus type	busbar	logical bus	bus

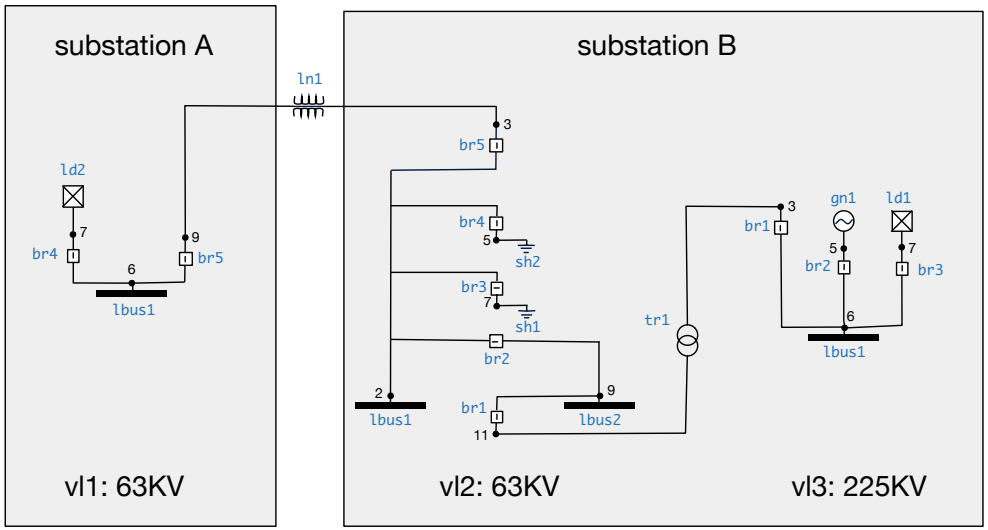
- GRG-NB and GRG-BB
  - topology optimization
  - NB: planning studies
  - BB: operational studies

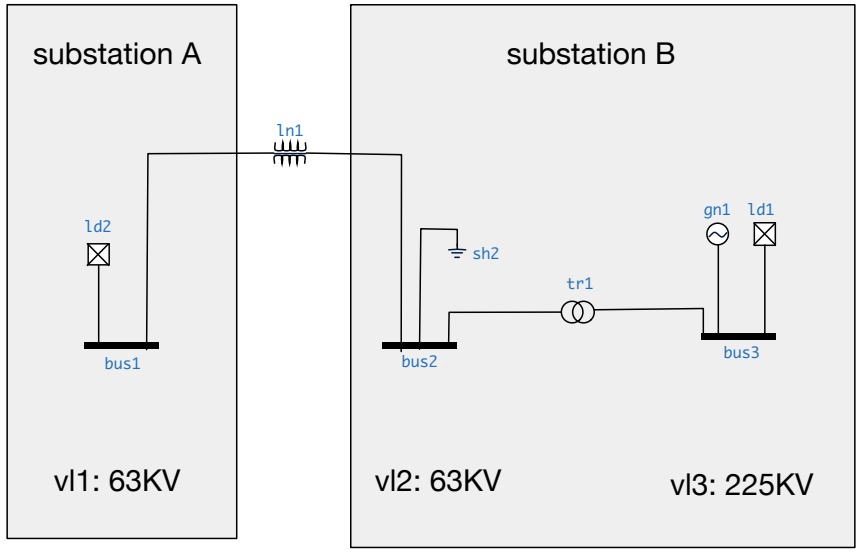
From GRG-ND to GRG-BB



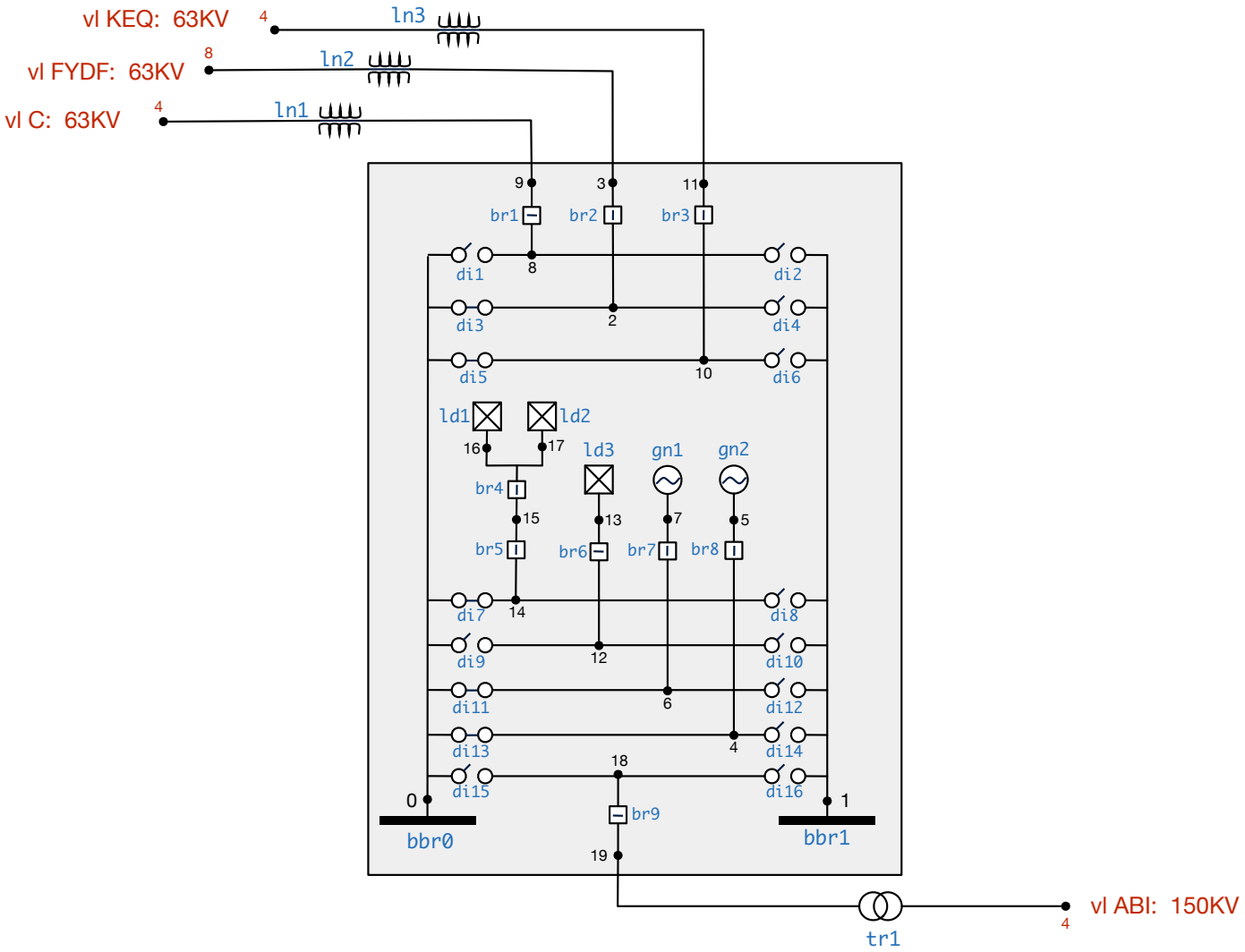


From GRG-BB to GRB-BBCH

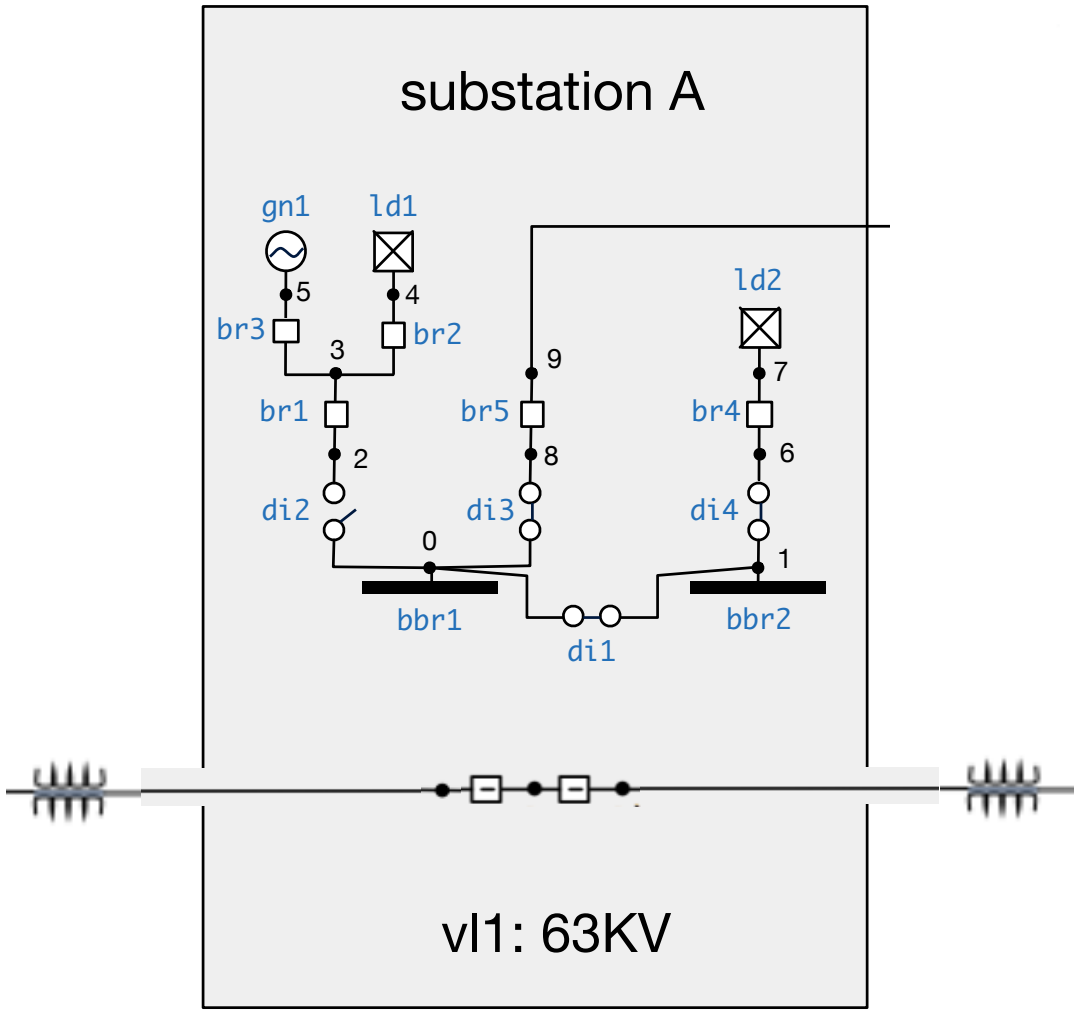




Real-Life Networks (at RTE)



Real-Life Networks (at RTE)



Key Concepts of the Format

- Electrical abstractions
  - e.g., admittance
- Component references
  - JSON pointers
- Explicit distinction between
  - parameters and decision variables
- Logical buses
  - abstraction of busbars in bus breaker representation
- Substations
  - with voltage levels

Variables versus Parameters

- Key motivation
  - one network description
  - multiple uses (opf, ots, ..)
- Assignments
  - specializes the networks
  - specific examples of transformations
- Basic philosophy
  - one physical network
  - separate sections describing configurations, setpoints, solutions, …


```
"line_2": {
  "angle_difference": {
    "var": {
      "lb": -0.5235987755982988,
      "ub": 0.5235987755982988
    }
  },
```

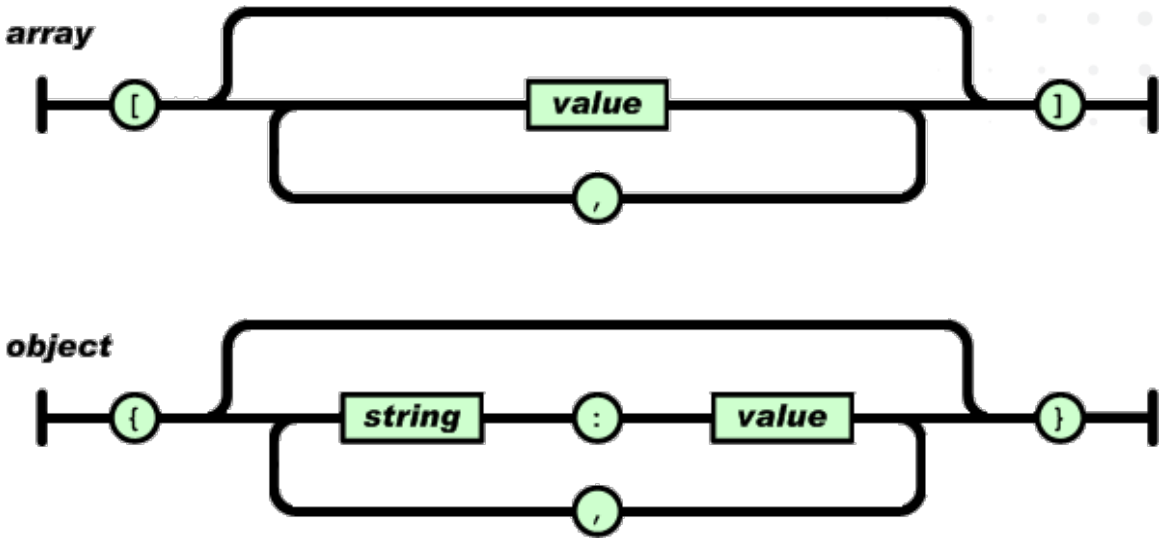
Overview

- The team
- Project objectives
- Technical approach
- ***Accomplishments***
  - format
- Goals
- Tech-to-Market
- Collaborations

JSON 101

- **Primitive types**
  - string
  - number (int or float)
  - boolean (true, false)
  - null

Recursive Types



Key benefit: JSON can be parsed into a dictionary-like data structure in any language

JSON 101 Examples

An array:

```
["item", 1, true, null, "item", 2.5]
```

An object:

```
{"item_1":10,
 "item_2":1,
 "item_5":null,
 "item_4":2.5}
```

A matrix:

```
[[1,2,3],
 [4,5,6],
 [7,8,9]
]
```

JSON Schema

- Specify the format formally
- Can we use to check whether a file is valid

```
{“required": ["link", "demand"],
 "properties": {
  "type":{
   "type":"string",
   "pattern":"load"
  },
  "link":{"type":"string"},
  "demand":{"$ref": “#/…”}
}
```

GRG Network


```
"network": {
    "type": "object",
    "required": ["type", "subtype", "per_unit", "flat", "units", "base_mva", "id", "description", "components"],
    "properties": {
    "id": {"type": "string"},
    "type": {"enum": ["network"]},
    "subtype": {"enum": ["node_breaker", "bus_breaker", "bus_branch"]},
    "description": {"type": "string"},
    "per_unit": {"type": "boolean"},
    "base_mva": {"type": "number"},
    "units": {
          "voltage": {"enum": ["volt", "V", "kilo_volt", "kV", "pu"]},
          "current": {"enum": ["ampere", "A", "kilo_ampere", "kA", "pu"]},
          "angle"  : {"enum": ["degree", "radian", "rad"]},
        “active_power": {"enum": ["watt", "W", "mega_watt", "MW", "pu"]},
          "reactive_power": {"enum": ["volt_ampere_reactive", "VAR", "mega_volt_ampre_reactive", "MVAR", "pu"]},
         "impedance": {"enum": ["ohm", "Omega", "pu"]},
        "resistance": {"enum": ["ohm", "Omega", "pu"]},
        "reactance": {"enum": ["ohm", "Omega", "pu"]},
        "conductance": {"enum": ["siemens", "S", "pu"]},
        "susceptance": {"enum": ["siemens", "S", “pu"]},
       }
   },
   "flat": {"type": "boolean"},
```

network properties

```
   "components": {
    "type": "object",
   "patternProperties": {
        ".*": {"$ref": "#/network_component"}
    }
  },
```

components’ description

```
  "assignments": {
      "$ref": “#/assignments"
    }
}
```

components’ assignments

GRG Network

```
{
  “grg-version”: “1.0.0”,
  “network”: {
     …
      “components”: {
         “substation_A”: {
            …
            “voltage_level_C” : {
               “GN1”: {
                     . . .
               },
         }
     },
  },
   “assignments”: {
       …
   }
},
  “mappings”: {
       . . .
  }
}
```

A GRG document is composed by:

- A “grg-version” field;
- A “network” object;
- A “mapping”object.

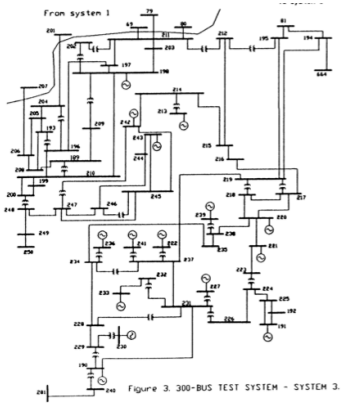
GRG Network

```
{
  “grg-version”: “1.0.0”,
  “network”: {
     …
      “components”: {
         “substation_A”: {
            …
            “voltage_level_C” : {
               “GN1”: {
                     . . .
               },
         }
     },
  },
   “assignments”: {
       …
   }
},
  “mappings”: {
       . . .
  }
}
```

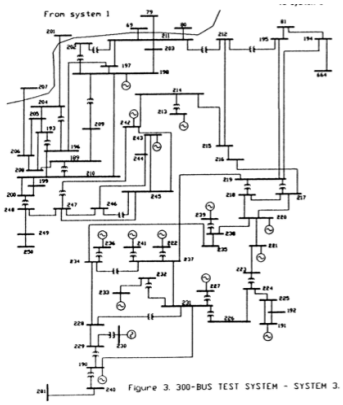
A GRG document is composed by:

- A “grg-version” field;
- A “network” object;
- A “mapping”object.

Networks

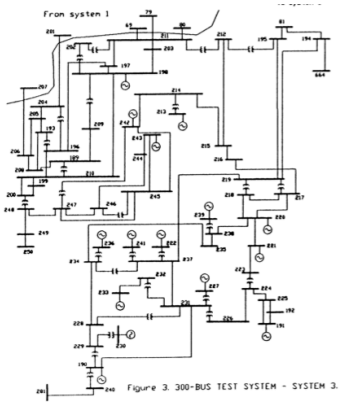


apply mapping



Assignment 1

apply mapping



Assignment 2

GRG Network: Assignment

Example

```
"assignment": {
  "switch_vl1_br1": {
      "status": "on"
   },
  "busbar_vl1_bbr1" : {
     "voltage" : {
         "angle" : 4.23,
         "magnitude" : 63.12
     }
  },
}
```

An “assignment” describes value assignments for network components’ objects.

GRG Network: Mappings

Example

```
mappings:{

”disconnectors_assignment" : {
   "components" : {
      "substation_sA" : {
         "substation_components": {
            "voltage_level_vl1" : {
               "voltage_level_components" {
                  "switch_vl1_di1" : {"status" : "on"},
                  "switch_vl1_di2" : {"status" :
                  “off"},
                  "switch_vl1_di3" : {"status" : “on"}
               }}}}},

”breakers_assignment" : {
   "components" : {
      "substation_sA" : {
         "substation_components": {
            "voltage_level_vl1" : {
               "voltage_level_components" {
                  "switch_vl1_br1" : {"status" : "on"},
                  "switch_vl1_br2" : {"status" : “on"},
                  "switch_vl1_br3" : {"status" : “off"}
               }}}}},
}
```

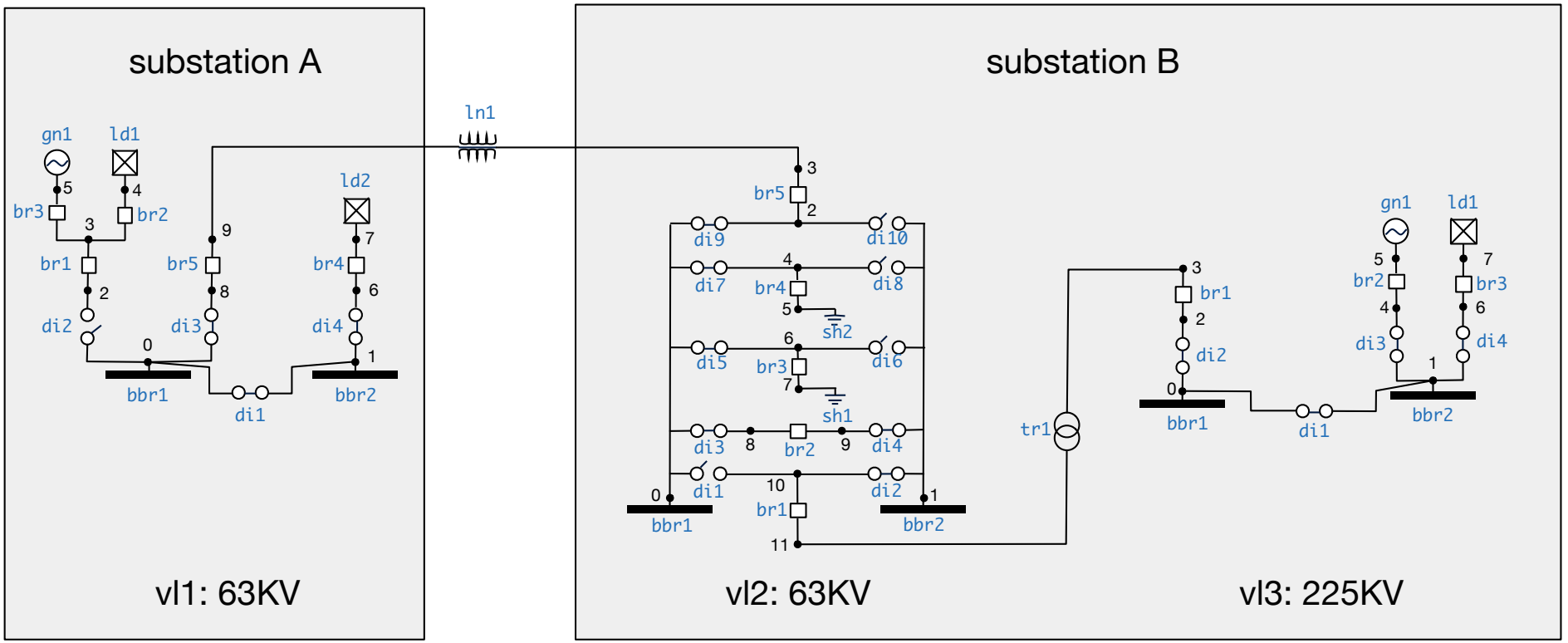
The network “mappings” describe sets of component assignments which can be used to define a network instance.

GRG Network: Hierarchy

Network

A GRG network is organized as:

- A set of substations
- A set of transmission lines

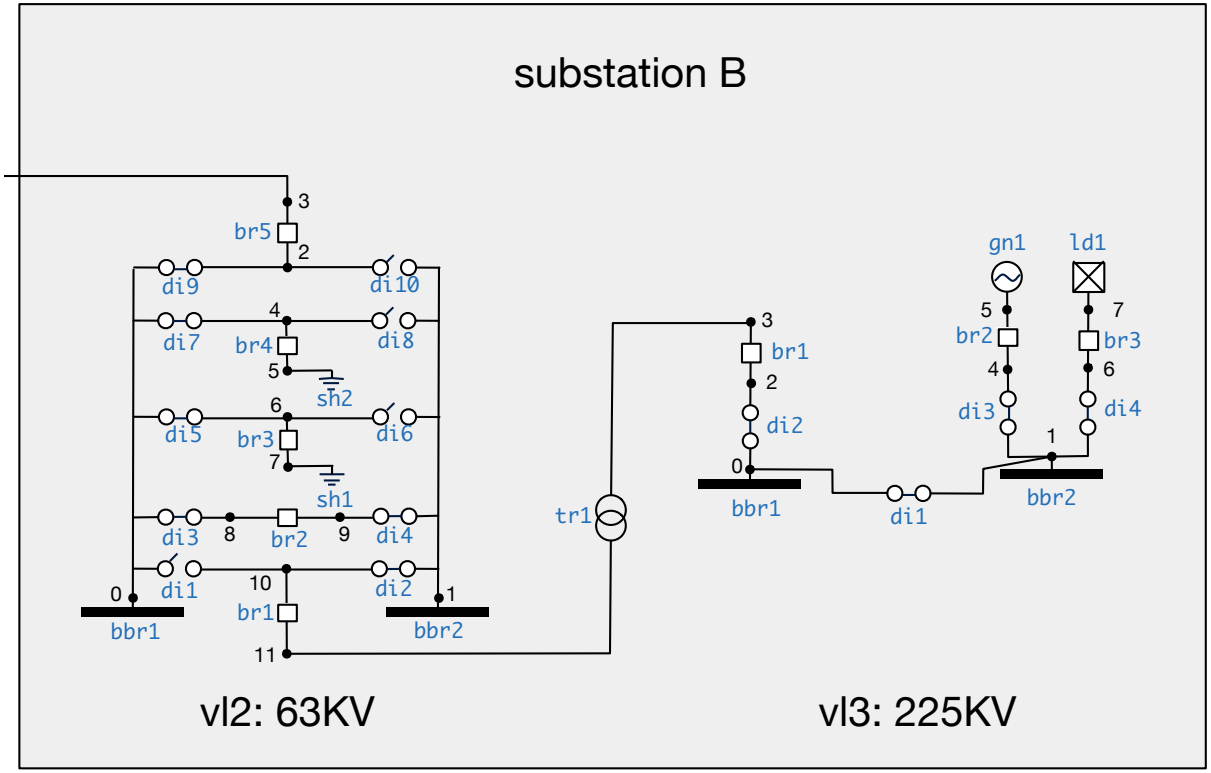


GRG Network: Hierarchy

Substation

A substation is a collection of:

- Voltage Levels
- Transformers

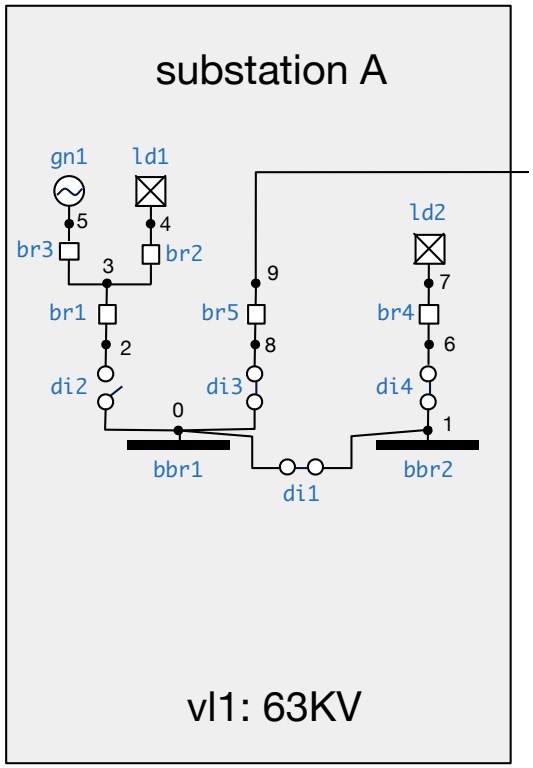


GRG Network: Hierarchy

Voltage Level

A voltage level is a collection of:

- Buses (busbars, logical buses, buses)
- Loads
- Synchronous Condensers
- Generators
- Switches
- Shunts



Electrical Values

Impedance

```
"impedance": {
  "required": ["resistance", "reactance"],
  "additionalProperties": false,
  "properties": {
  "resistance": {"$ref": "#/values/abstract_value"},
  "reactance": {"$ref": "#/values/abstract_value"}
  }
}
```

$$Z = R + jX$$

Example

```
“impedance_1” : {
    “resistance” : 6.52,
    “reactance” : 2.39
}
```

GRG name	Symbol	Unit
impedance	Z	Ohm (Ω)
resistance	R	Ohm (Ω)
reactance	X	Ohm (Ω)

Electrical Values

Current Limits

```
"current_limits": {
  "type": "array",
  "items": {
  "type": "object",
  "properties": {
  "min": {"$ref": "#/values/extended_number"},
  "max": {"$ref": "#/values/extended_number"},
  "report": {
   "enum": ["off", "on"]
  },
  "duration": {"$ref": "#/values/extended_number"}
  }
  }
}
```

Amount of current: $[I^{min}, I^{max})$ the branch can safely carry for a duration d.

GRG name	Symbol	Unit
min	I^{min}	kiloAmpere (kA)
max	I^{max}	kiloAmpere (kA)
duration	d	Minutes (min)

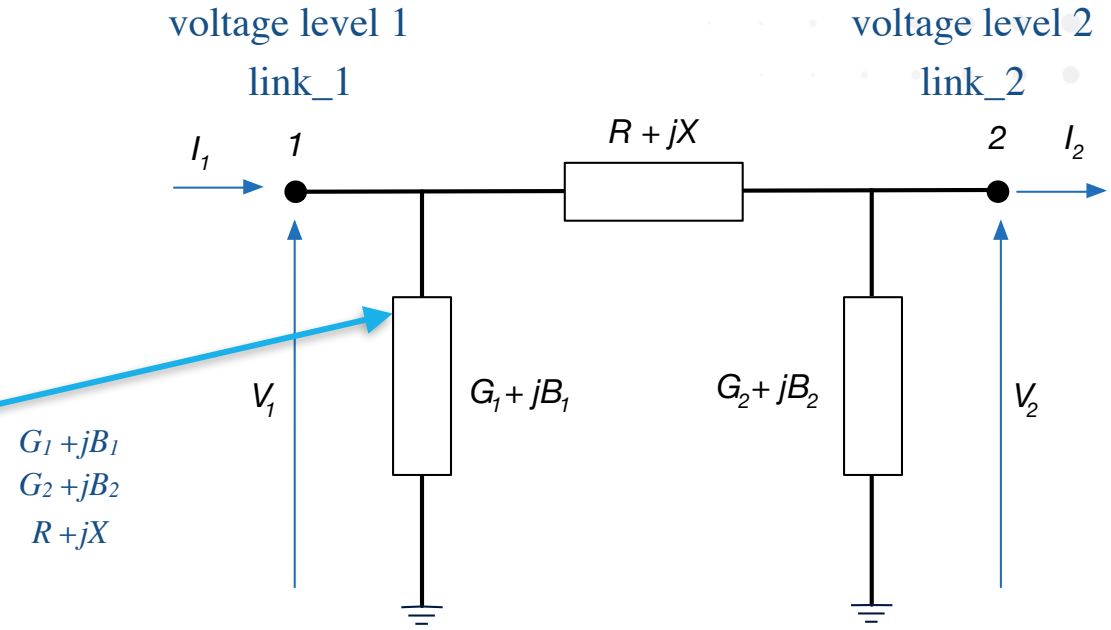
Example

```
"current_limits_1" : [
   { "duration" : “Inf", "max" : 563,   "min" : 0,   "report" : “off" },
   { "duration" : “Inf", "max" : 746,   "min" : 563, "report" : “on”},
   { "duration" : 6300,  "max" : “Inf", "min" : 746, "report" : “off”}
],
```

Network Components

AC Line

```
"AC_line": {
   "properties": {
  "type": {"enum": ["ac_line"]},
  "id": {"type": "string"},
  "description": {"type": "string"},
  "link_1": {"type": "string"},
  "link_2": {"type": "string"},
  "voltage_level_1_id": {"type": "string"},
  "voltage_level_2_id": {"type": "string"},
  "shunt_1": {"$ref": "#/admittance_rectangular"},
  "shunt_2": {"$ref": "#/admittance_rectangular"},
  "impedance": {"$ref": "#/impedance_rectangular"},
  "angle_difference": {"$ref": "#/abstract_value"},
  "current_limits_1": {"$ref": "#/current_limits"},
  "current_limits_2": {"$ref": "#/current_limits"}
  }
}
```



$$I_1 = Y_1 \cdot V_1 + \frac{1}{Z}(V_1 - V2)$$

$$I_2 = Y_2 \cdot V_2 + \frac{1}{Z}(V_1 - V2)$$

$$S_i = \bar{I}_i \cdot V_i \qquad (i \in \{1, 2\})$$

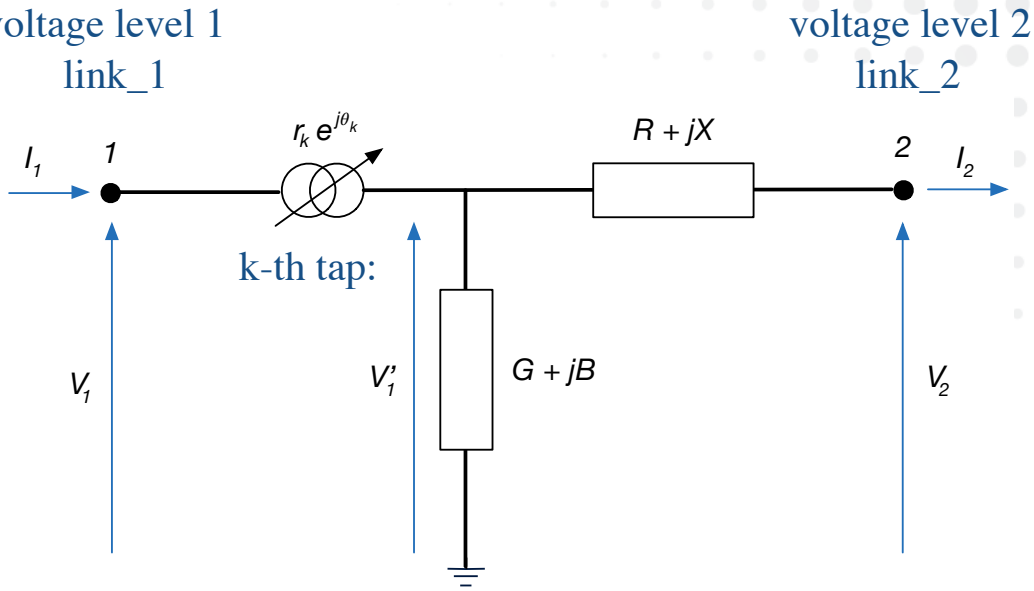
Network Components

Two Windings Transformer

```
"two_windings_transformer": {
    "type": {"enum": [“line"]},
    "id": {"type": “string"},
    "description": {"type": “string"},
    "link_1": {"type": “string"},
    "link_2": {"type": “string"},
    "voltage_level_1_id": {"type": “string"},
    "voltage_level_2_id": {"type": “string"},
    "shunt": {"$ref": “#/admittance”},
    "impedance": {"$ref": “#/impedance"},
    "angle_difference": {"$ref": “#/abstract_value"},
    "current_limits_1": {"$ref": “#/current_limits"},
    "current_limits_2": {"$ref": “#/current_limits"},
    "tap_changer": {
        "position": {"$ref": “#/abstract_value"},
        "OLTC": {
            "status": {"$ref": “#/status"},
            "regulating_side": {"enum": [1, 2]},
            "target_voltage": {"$ref": “#/values/abstract_value"}
        }
    },
    "steps": {
        "type": “array",
            "items": {
                "impedance": {"$ref": “#/impedance"},
                "shunt": {"$ref": “#/admittance"},
                "tap_ratio": {"$ref": “#/abstract_value"},
                "angle_shift": {"$ref": “#/abstract_value”}
            }
        }
    }
}
```

$G + jB$
$R + jX$

$R_k + jX_k$
$G_k + jB_k$
r_k
θ_k



$$I_1' = \rho_k \cdot (Y_1 \cdot V_1' + 1/Z_K (V_1' - V_2)$$

$$I_1' = 1/Z_K (V_2 - V_1')$$

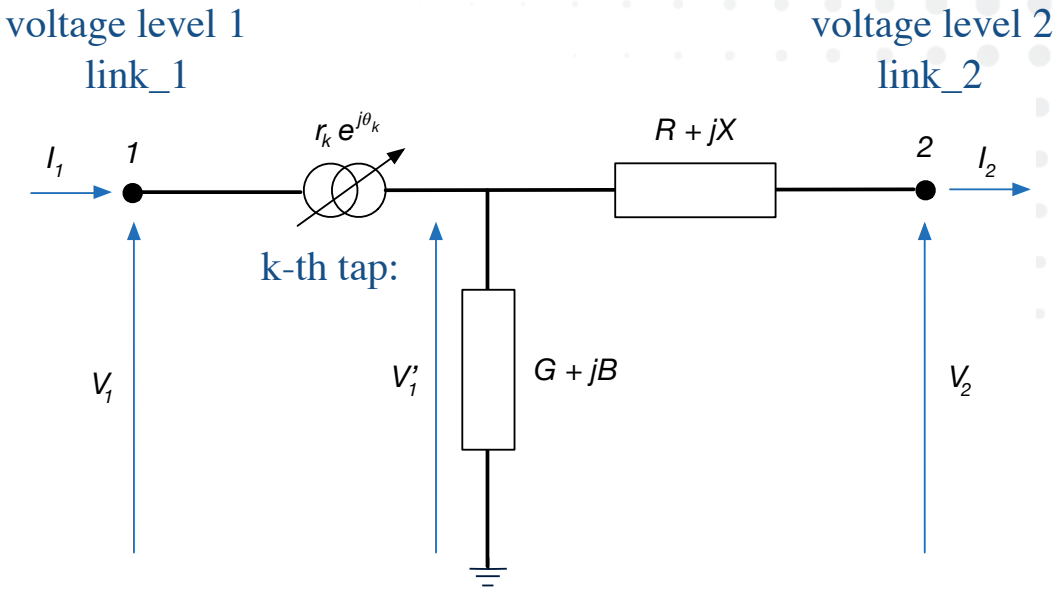
$$V_1' = \rho_k \cdot V_1$$

$$S_i = \bar{I}_i \cdot V_i \qquad (i \in \{1, 2\})$$

Network Components

Example

```
"two_windings_transformer_tr1" : {
    "type" : “two_windings_transformer”,
    "id" : “tr1”,
    "link_1" : “11”,
    "link_2" : “3”,
    "voltage_level_1_id" : “vl2”,
    "voltage_level_2_id" : “vl3”,
    "angle_difference" : {"var": {"lb": -30.0, "ub": 30.0}},
    "current_limits_1" : [
        {"duration":"Inf", “max”:1029, “min”:0, “report”:“off"},
        {"duration":1200, “max”:1342, “min”:1029, “report":"off"},
        {“duration":300, “max”:1790, "min":1342, “report":"off"}
        {"duration":60, “max”:"Inf", "min":1790, “report”:"off"}
    ],
    "impedance": {"reactance": 60.1, "resistance": 1.512},
    "shunt": {"conductance":0, "susceptance": 0},
    "tap_changer" : {
        "OLTC" : {
            "regulating_side": 1,
            "status": {"var":["off", “on"]},
            "target_voltage": {“var":{"lb":"-Inf", “ub":"Inf"}}
        },
        "position": {"var": {"lb":0, "ub":2}},
        “steps”:[{
            "impedance" : {"reactance" : 0, "resistance" : 0},
            "shunt" : {"conductance" : 0, "susceptance" : 0},
            "tap_ratio" : 1.011, "angle_shift": 0
            }, {
            "impedance" : {"reactance" : 0, "resistance" : 0},
            "shunt" : {"conductance" : 0, "susceptance" : 0},
            "tap_ratio" : 1.0, ”angle_shift" : 0,
            },{
            "impedance" : {"reactance" : 0, "resistance" : 0},
            "shunt" : {"conductance" : 0, "susceptance" : 0},
            "tap_ratio" : 0.988, "angle_shift" : 0 }]
  }
}
```



$$I_1' = \rho_k \cdot (Y_1 \cdot V_1' + 1/Z_K (V_1' - V_2)$$

$$I_1' = 1/Z_K (V_2 - V_1')$$

$$V_1' = \rho_k \cdot V_1$$

$$S_i = \bar{I}_i \cdot V_i \qquad (i \in \{1, 2\})$$

Network Components

Generator

```
"generator": {
    "type": {"enum": [“generator"]},
    "subtype": {"enum": ["hydro", "wind", "thermal", "other", "nuclear", “solar"]},
    "id": {"type": “string"},
    "link": {"type": “string"},
    "voltage_regulation": {"$ref": “#/abstract_status"},
    "startup_cost": {"type": “number"},
    "shutdown_cost": {"type": “number"},
    "output": {"$ref": “#/power"},
    "PQ_curve": {
        "type": “array", "items": {
            "active": {"type": “number"},
            "reactive": {
                "upper_limit": {"type": “number"},
                "lower_limit": {"type": “number"}}
        }
    }
}
```

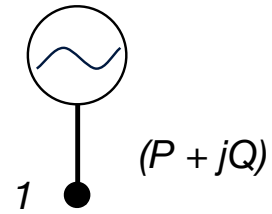
$P + jQ$

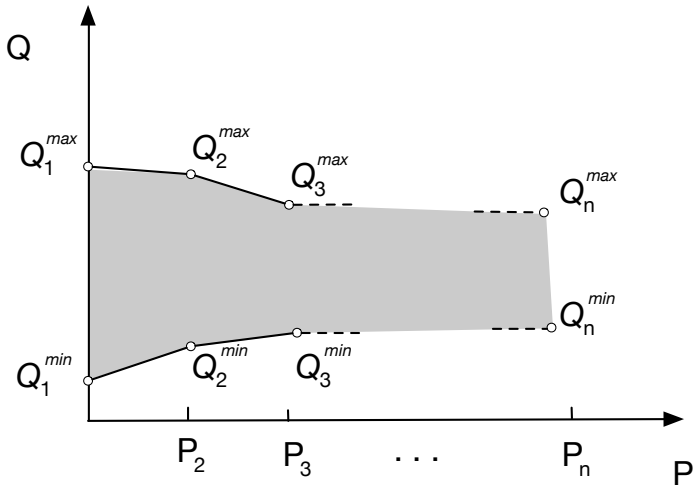
P

Q^{min}

Q^{max}

```
"generator_vl1_gn1" : {
    "type" : “generator”,
    "subtype" : “solar",
    "id" : “gn1",
    "link" : “5",
    "voltage_regulation" : “off",
    "PQ_curve" : [
        {"active":0, "reactive":{"lower_limit":0, "upper_limit":0}},
       {"active" : 20.7, "reactive":{"lower_limit":0, “upper_limit”:0}}],
    "output" : {
        “active": {"var": {"lb":0, "ub": 20.7}},
        "reactive": {"var": {"lb":0, "ub": 0.0 }}
  }
}
```





b.

Overview

- The team
- Project objectives
- Technical approach
- ***Accomplishments***
  - transformation and scripting
- Goals
- Tech-to-Market
- Collaborations

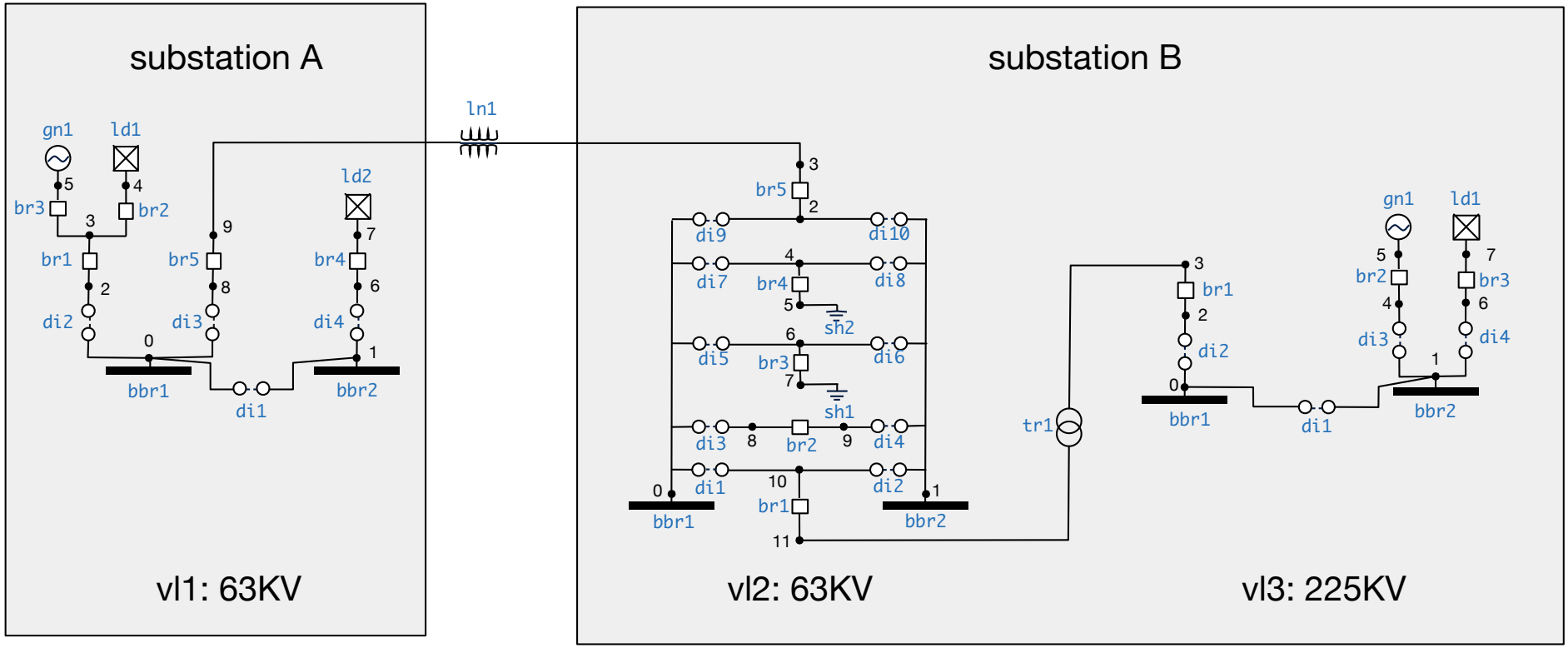
GRG Scripting Language

```
{
   "grid_version": “0.1.0a0",
   "script": {
      "scope": “/…/input_networks/”,
      "network_filename": “demo_net”,
      "steps": [
         {
            "name": “net_0”,
            "type": “load",
            "status": “on”
         },
         {
            "name": “net_1",
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "disconnectors_assignment"
         },
         {
            "name": “net_2”,
            "type": “transform",
            "subtype": “nodebreaker_to_busbreaker",
            "status": “on"
         },
         {
            "name": “net_3",
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "breakers_assignment"
         },
         {
            "name": “net_4”,
            "type": “transform",
            "subtype": “busbreaker_to_busbranch",
            "status": “on"
         },
         {
            "name": “net_5”,
            "type": “transform",
            “subtype”: “per_unit”,
            “nominal_power”: 100.00,
            "status": “on"
         },
         {
            "name": “net_6”,
            "type": “solve",
            "subtype": “ac_opf”,
            "status": “on”,
            "solver": “ipopt",
            "relaxation": “ACPOL”,
            “starting_points”: {“apply_assignment”:
            “starting_points"}
         },
         {
            “name”: “net_7”,
            "type": “rollback",
            "target": “net_0“,
            "status": “on”
         },
         {
            "type": “output”,
            “subtype”: “network”,
            "status": “on"
            “file_out”: “/…/output_networks/demo_net”
         }
      ]
   }
}
```

GRG Script: Load

```
{
   "grid_version": “0.1.0a0",
   "script": {
      "scope": “/…/input_networks/”,
      "network_filename": “demo_net”,
      "steps": [
         {
            "name": “net_0”,
            "type": “load",
            "status": “on”
         },
         {
            "name": “net_1",
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "disconnectors_assignment"
         },
         {
            "name": “net_2”,
            "type": “transform",
            "subtype": “nodebreaker_to_busbreaker",
            "status": “on"
         },
         {
            "name": “net_3",
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "breakers_assignment"
         },
         {
            "name": “net_4”,
            "type": “transform",
            "subtype": “busbreaker_to_busbranch",
            "status": “on"
         },
         {
            "name": “net_5”,
            "type": “transform",
            “subtype”: “per_unit”,
            “nominal_power”: 100.00,
            "status": “on"
         },
         {
            "name": “net_6”,
            "type": “solve",
            "subtype": “ac_opf”,
            "status": “on”,
            "solver": “ipopt",
            "relaxation": “ACPOL”,
            “starting_points”: {“apply_assignment”:
            “starting_points"}
         },
         {
            “name”: “net_7”,
            "type": “rollback",
            "target": “net_0“,
            "status": “on”
         },
         {
            "type": “output”,
            “subtype”: “network”,
            "status": “on"
            “file_out”: “/…/output_networks/demo_net”
         }
      ]
   }
}
```

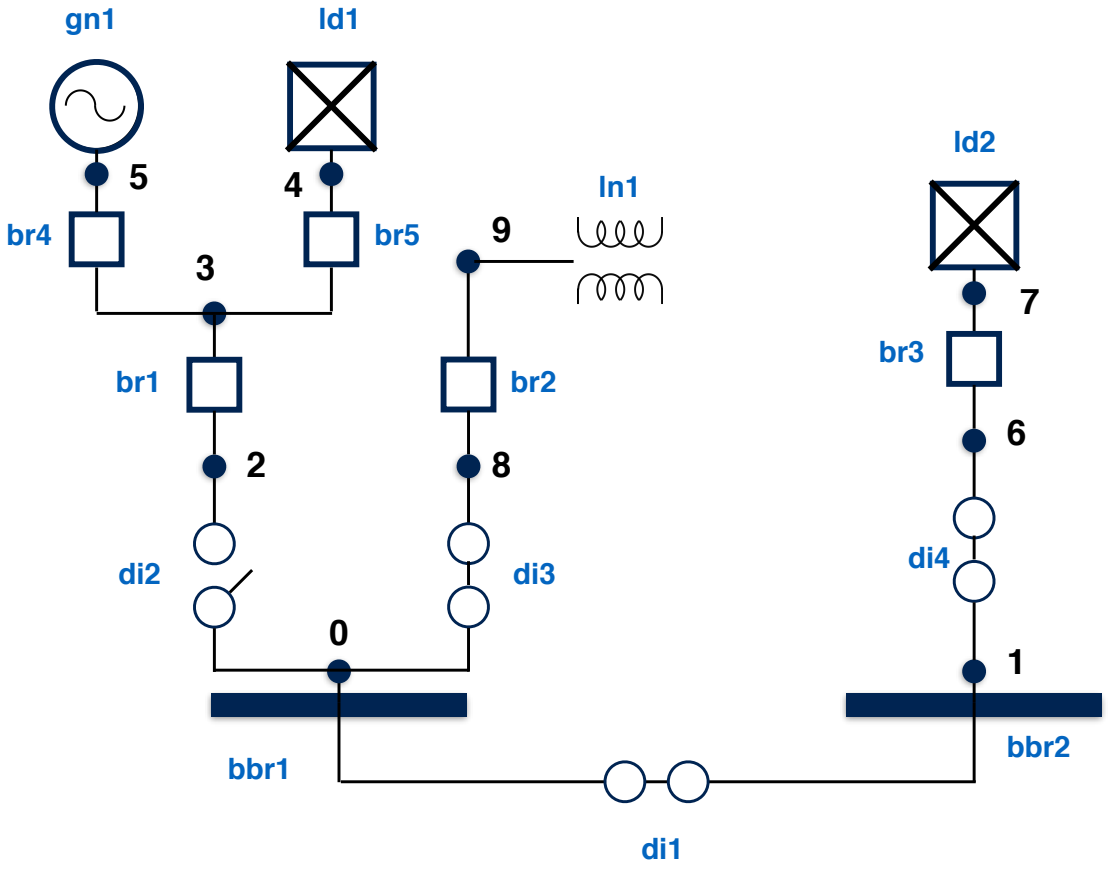
GRG Script: Load



GRG Script: Assignment

```
{
   "grid_version": “0.1.0a0",
   "script": {
      "scope": “/…/input_networks/”,
      "network_filename": “demo_net”,
      "steps": [
         {
            "name": “net_0”,
            "type": “load",
            "status": “on”
         },
         {
            "name": “net_1",
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "disconnectors_assignment"
         },
         {
            "name": “net_2”,
            "type": “transform",
            "subtype": “nodebreaker_to_busbreaker",
            "status": “on"
         },
         {
            "name": “net_3”,
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "breakers_assignment"
         },
         {
            "name": “net_4”,
            "type": “transform",
            "subtype": “busbreaker_to_busbranch",
            "status": “on"
         },
         {
            "name": “net_5”,
            "type": “transform",
            “subtype”: “per_unit”,
            “nominal_power”: 100.00,
            "status": “on"
         },
         {
            "name": “net_6”,
            "type": “solve",
            "subtype": “ac_opf”,
            "status": “on”,
            "solver": “ipopt",
            "relaxation": “ACPOL”,
            “starting_points”: {“apply_assignment”:
            “starting_points"}
         },
         {
            “name”: “net_7”,
            "type": “rollback",
            "target": “net_0“,
            "status": “on”
         },
         {
            "type": “output”,
            “subtype”: “network”,
            "status": “on"
            “file_out”: “/…/output_networks/demo_net”
         }
      ]
   }
}
```

GRG Script: Assignment



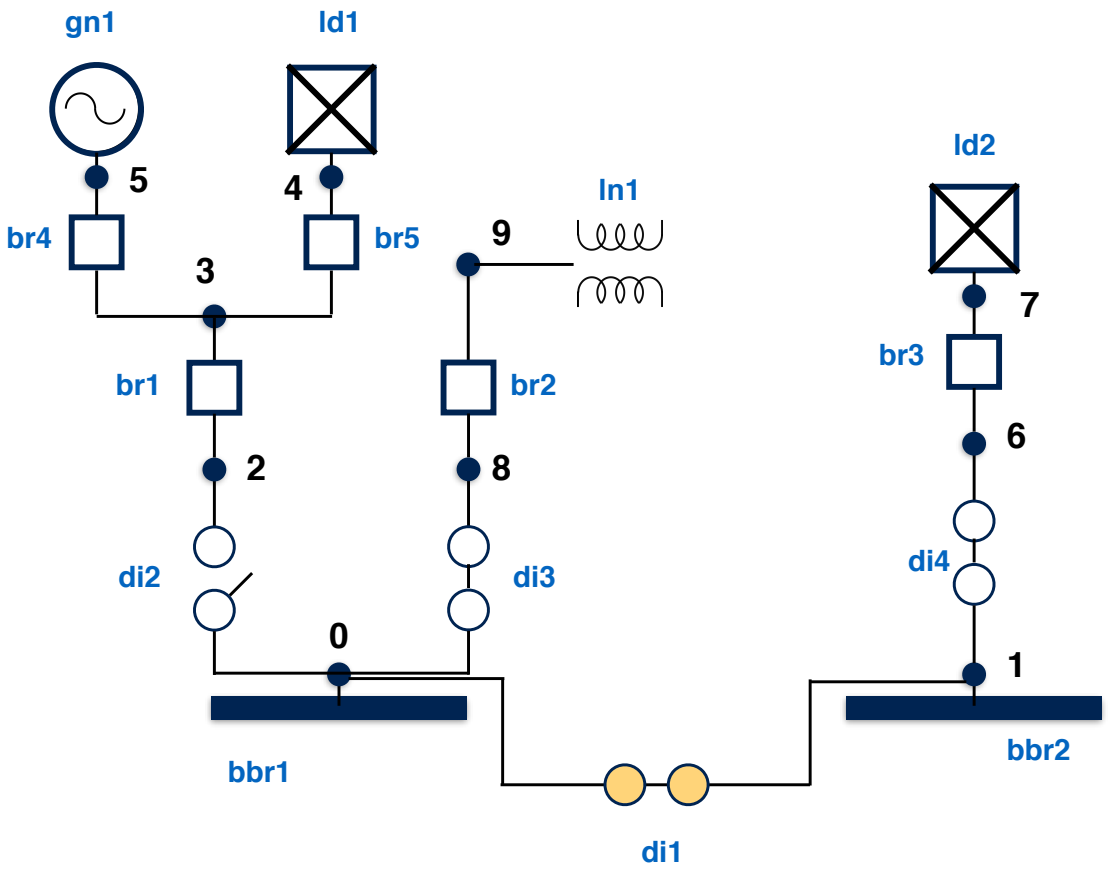

```
"voltage_level_vl1" :
{
  "id" : "vl1",
  "type" : "voltage_level",
  "voltage" : {
    "lower_limit" : 59.0,
    "nominal_value" : 63.0,
    "upper_limit" : 68.0
  },
  "voltage_level_components" :
  {
    “vl1_bbr1” : {“link”: 0, … },
    “vl1_bbr2” : {“link”: 1, … },
    "vl1_gn1" : {“link”: 5, …},
    "vl1_ld1" : {“link”: 4, …},
    "vl1_ld2" : {“link”: 7, …},
    "vl1_br1" : {“link_1”: 3, “link_2”: 2, …},
    "vl1_br2" : {“link_1”: 9, “link_2”: 8, …},
    "vl1_br3" : {“link_1”: 7, “link_2”: 6, …},
    "vl1_br4" : {“link_1”: 5, “link_2”: 3, …},
    "vl1_br5" : {“link_1”: 4, “link_2”: 3, …},
    "vl1_di1" : {“link_1”: 0, “link_2”: 1, …},
    "vl1_di2" : {“link_1”: 2, “link_2”: 0, …},
    "vl1_di3" : {“link_1”: 8, “link_2”: 0, …},
    "vl1_di4" : {“link_1”: 6, “link_2”: 1, …},
  }
},

assignments: {
..“voltage_level_vl1”: {
   “voltage_level_components”:
    {
    "vl1_di1": { status = “on”},
    "vl1_di2": { status = “off”},
    "vl1_di3": { status = “on”},
    "vl1_di4": { status = “on"}
    }
 }
}
```

GRG Script: Node-breaker To Bus-breaker

```
{
   "grid_version": “0.1.0a0",
   "script": {
      "scope": “/…/input_networks/”,
      "network_filename": “demo_net”,
      "steps": [
         {
            "name": “net_0”,
            "type": “load",
            "status": “on”
         },
         {
            "name": “net_1",
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "disconnectors_assignment"
         },
         {
            "name": “net_2”,
            "type": “transform",
            "subtype": “nodebreaker_to_busbreaker",
            "status": “on"
         },
         {
            "name": “net_3",
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "breakers_assignment"
         },
         {
            "name": “net_4”,
            "type": “transform",
            "subtype": “busbreaker_to_busbranch",
            "status": “on"
         },
         {
            "name": “net_5”,
            "type": “transform",
            “subtype”: “per_unit”,
            “nominal_power”: 100.00,
            "status": “on"
         },
         {
            "name": “net_6”,
            "type": “solve",
            "subtype": “ac_opf”,
            "status": “on”,
            "solver": “ipopt",
            "relaxation": “ACPOL”,
            “starting_points”: {“apply_assignment”:
            “starting_points"}
         },
         {
            “name”: “net_7”,
            "type": “rollback",
            "target": “net_0“,
            "status": “on”
         },
         {
            "type": “output”,
            “subtype”: “network”,
            "status": “on"
            “file_out”: “/…/output_networks/demo_net”
         }
      ]
   }
}
```

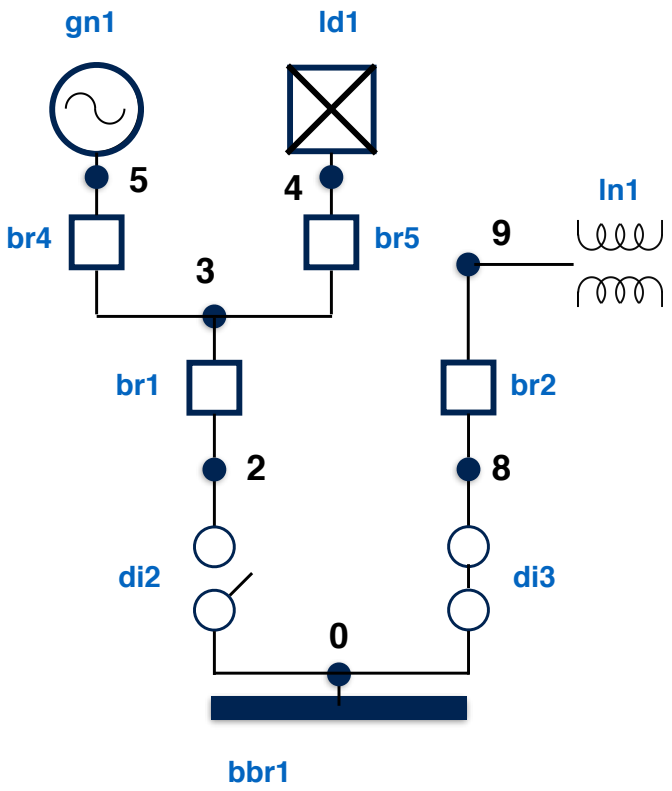
GRG Script: Node-breaker To Bus-breaker

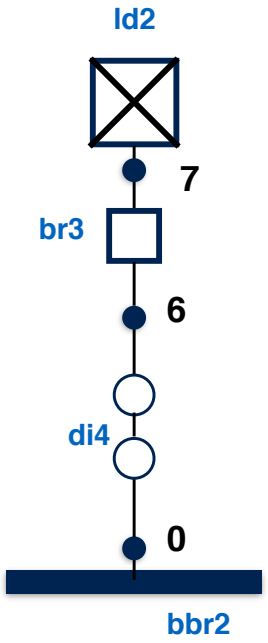



```
"voltage_level_vl1" :
{
  "id" : "vl1",
  "type" : "voltage_level",
  "voltage" : {
    "lower_limit" : 59.0,
    "nominal_value" : 63.0,
    "upper_limit" : 68.0
  },
  "voltage_level_components" :
  {
    “vl1_bbr1” : {“link”: 0, … },
    “vl1_bbr2” : {“link”: 1, … },
    "vl1_gn1" : {“link”: 5, …},
    "vl1_ld1" : {“link”: 4, …},
    "vl1_ld2" : {“link”: 7, …},
    "vl1_br1" : {“link_1”: 3, “link_2”: 2, …},
    "vl1_br2" : {“link_1”: 9, “link_2”: 8, …},
    "vl1_br3" : {“link_1”: 7, “link_2”: 6, …},
    "vl1_br4" : {“link_1”: 5, “link_2”: 3, …},
    "vl1_br5" : {“link_1”: 4, “link_2”: 3, …},
    "vl1_di1" : {“link_1”: 0, “link_2”: 1, …},
    "vl1_di2" : {“link_1”: 2, “link_2”: 0, …},
    "vl1_di3" : {“link_1”: 8, “link_2”: 0, …},
    "vl1_di4" : {“link_1”: 6, “link_2”: 1, …},
  }
},

assignments: {
..“voltage_level_vl1”: {
   “voltage_level_components”:
    {
    "vl1_di1": { status = “on”},
    "vl1_di2": { status = “off”},
    "vl1_di3": { status = “on”},
    "vl1_di4": { status = “on"}
    }
 }
}
```

GRG Script: Node-breaker To Bus-breaker

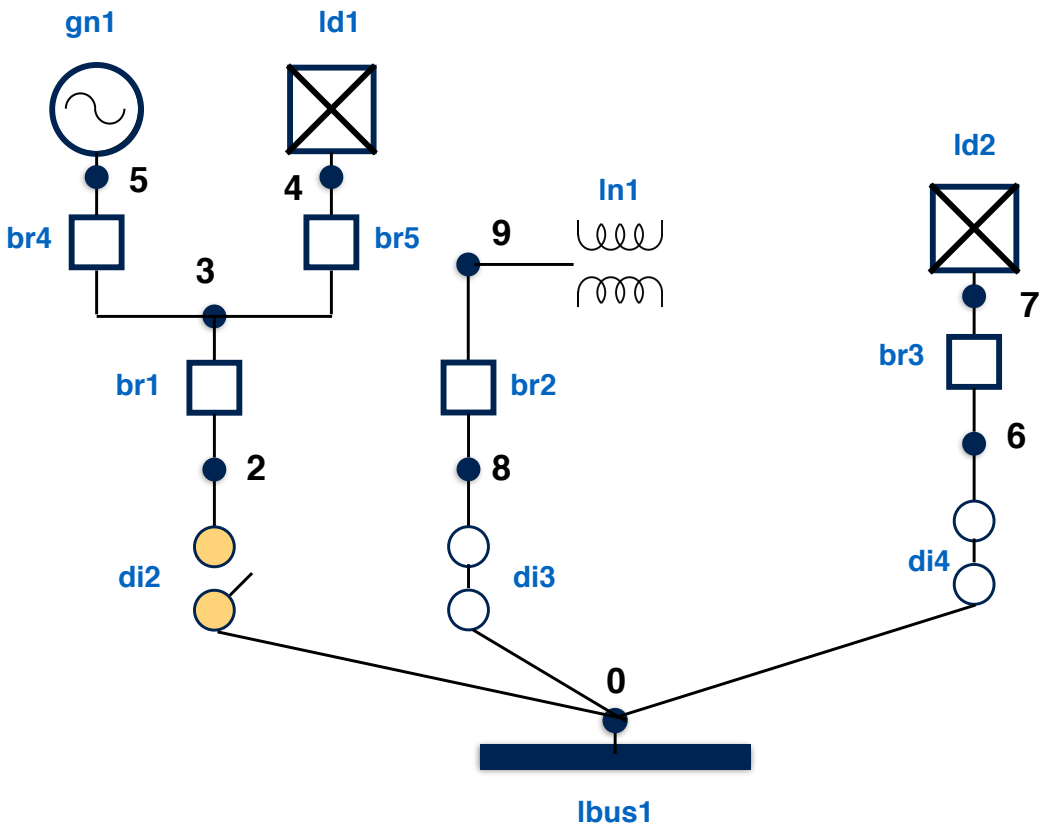





```
"voltage_level_vl1" :
{
  "id" : "vl1",
  "type" : "voltage_level",
  "voltage" : {
    "lower_limit" : 59.0,
    "nominal_value" : 63.0,
    "upper_limit" : 68.0
  },
  "voltage_level_components" :
  {
    “vl1_bbr1” : {“link”: 0, … },
    “vl1_bbr2” : {“link”: 0, … },
    "vl1_gn1" : {“link”: 5, …},
    "vl1_ld1" : {“link”: 4, …},
    "vl1_ld2" : {“link”: 7, …},
    "vl1_br1" : {“link_1”: 3, “link_2”: 2, …},
    "vl1_br2" : {“link_1”: 9, “link_2”: 8, …},
    "vl1_br3" : {“link_1”: 7, “link_2”: 6, …},
    "vl1_br4" : {“link_1”: 5, “link_2”: 3, …},
    "vl1_br5" : {“link_1”: 4, “link_2”: 3, …},
    "vl1_di2" : {“link_1”: 2, “link_2”: 0, …},
    "vl1_di3" : {“link_1”: 8, “link_2”: 0, …},
    "vl1_di4" : {“link_1”: 6, “link_2”: 0, …},
  }
},

assignments: {
..“voltage_level_vl1”: {
   “voltage_level_components”:
    {
    "vl1_di2": { status = “off”},
    "vl1_di3": { status = “on”},
    "vl1_di4": { status = “on"}
    }
 }
}
```

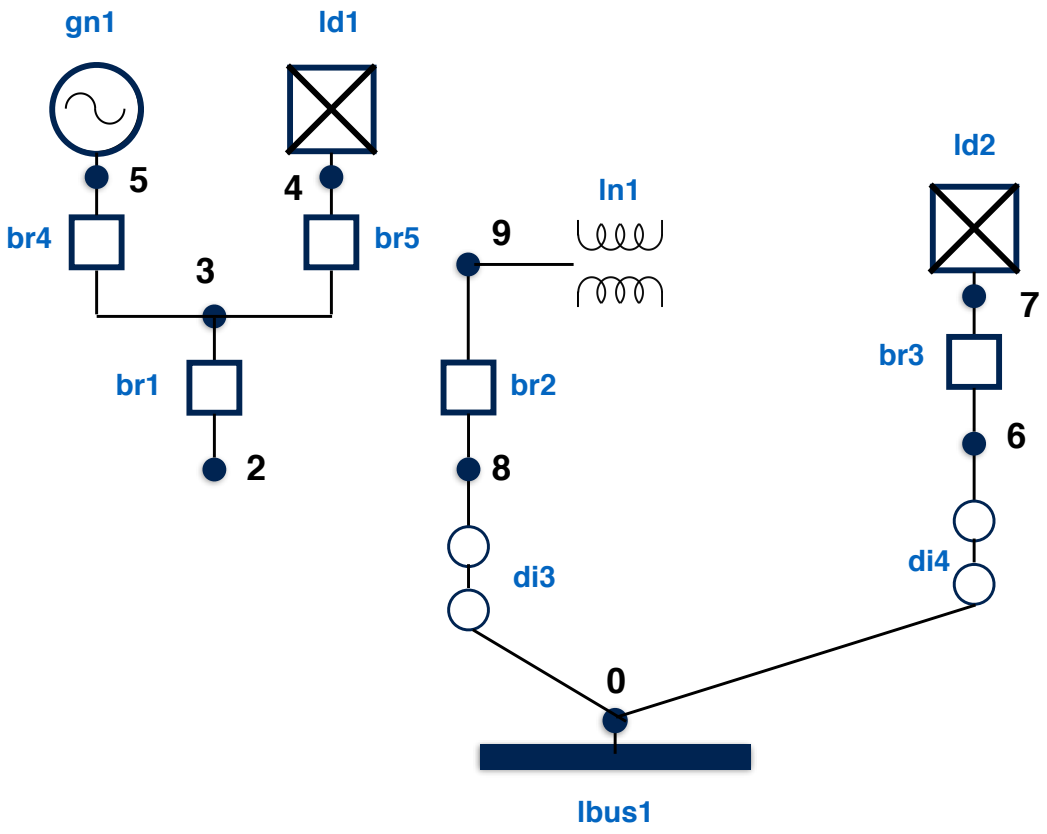
GRG Script: Node-breaker To Bus-breaker




```
"voltage_level_vl1" :
{
  "id" : "vl1",
  "type" : "voltage_level",
  "voltage" : {
    "lower_limit" : 59.0,
    "nominal_value" : 63.0,
    "upper_limit" : 68.0
  },
  "voltage_level_components" :
  {
    “vl1_lbus1” : {“link”: 0, … },
    "vl1_gn1" : {“link”: 5, …},
    "vl1_ld1" : {“link”: 4, …},
    "vl1_ld2" : {“link”: 7, …},
    "vl1_br1" : {“link_1”: 3, “link_2”: 2, …},
    "vl1_br2" : {“link_1”: 9, “link_2”: 8, …},
    "vl1_br3" : {“link_1”: 7, “link_2”: 6, …},
    "vl1_br4" : {“link_1”: 5, “link_2”: 3, …},
    "vl1_br5" : {“link_1”: 4, “link_2”: 3, …},
    "vl1_di2" : {“link_1”: 2, “link_2”: 0, …},
    "vl1_di3" : {“link_1”: 8, “link_2”: 0, …},
    "vl1_di4" : {“link_1”: 6, “link_2”: 0, …},
  }
},

assignments: {
..“voltage_level_vl1”: {
   “voltage_level_components”:
    {
    "vl1_di2": { status = “off”},
    "vl1_di3": { status = “on”},
    "vl1_di4": { status = “on”}
    }
 }
}
```

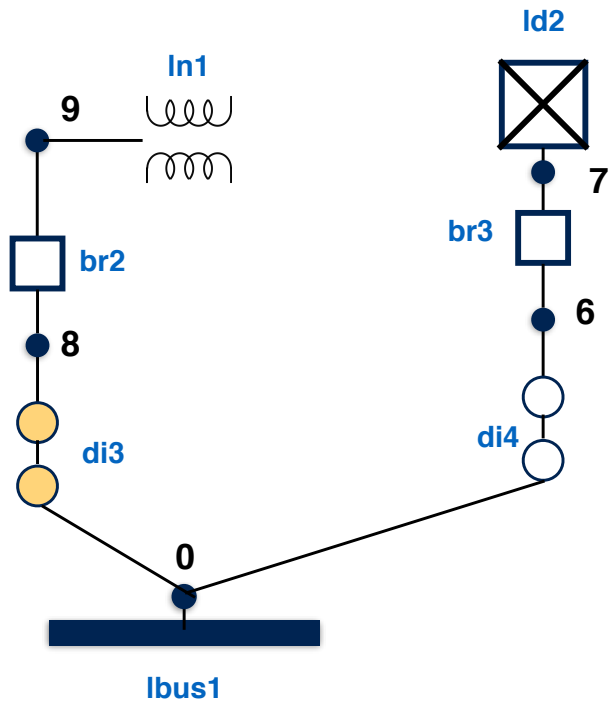
GRG Script: Node-breaker To Bus-breaker




```
"voltage_level_vl1" :
{
  "id" : "vl1",
  "type" : "voltage_level",
  "voltage" : {
    "lower_limit" : 59.0,
    "nominal_value" : 63.0,
    "upper_limit" : 68.0
  },
  "voltage_level_components" :
  {
    “vl1_lbus1” : {“link”: 0, … },
    "vl1_gn1" : {“link”: 5, …},
    "vl1_ld1" : {“link”: 4, …},
    "vl1_ld2" : {“link”: 7, …},
    "vl1_br1" : {“link_1”: 3, “link_2”: 2, …},
    "vl1_br2" : {“link_1”: 9, “link_2”: 8, …},
    "vl1_br3" : {“link_1”: 7, “link_2”: 6, …},
    "vl1_br4" : {“link_1”: 5, “link_2”: 3, …},
    "vl1_br5" : {“link_1”: 4, “link_2”: 3, …},
    "vl1_di3" : {“link_1”: 8, “link_2”: 0, …},
    "vl1_di4" : {“link_1”: 6, “link_2”: 0, …},
  }
},

assignments: {
..“voltage_level_vl1”: {
   “voltage_level_components”:
    {
    "vl1_di3": { status = “on”},
    "vl1_di4": { status = “on"}
    }
 }
}
```

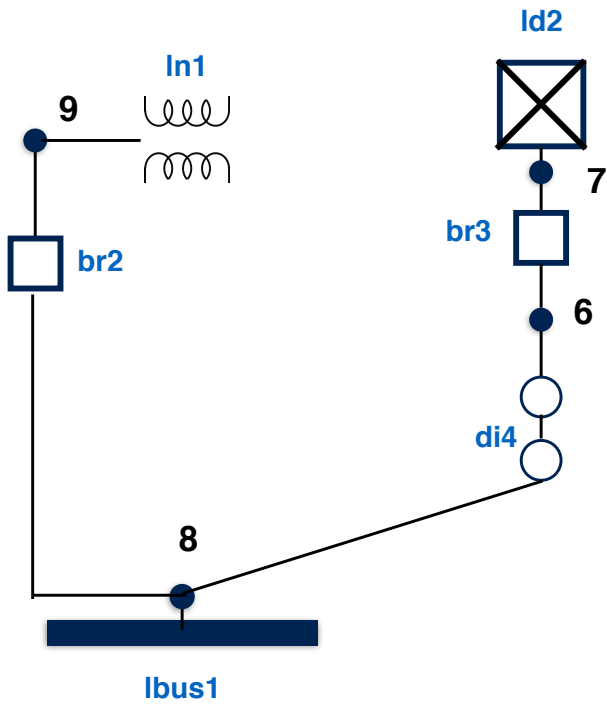
GRG Script: Node-breaker To Bus-breaker




```
"voltage_level_vl1" :
{
  "id" : "vl1",
  "type" : "voltage_level",
  "voltage" : {
    "lower_limit" : 59.0,
    "nominal_value" : 63.0,
    "upper_limit" : 68.0
  },
  "voltage_level_components" :
  {
    "vl1_lbus1" : {"link": 0, … },
    "vl1_ld2" : {"link": 7, …},
    "vl1_br2" : {"link_1": 9, "link_2": 8, …},
    "vl1_br3" : {"link_1": 7, "link_2": 6, …},
    "vl1_di3" : {"link_1": 8, "link_2": 0, …},
    "vl1_di4" : {"link_1": 6, "link_2": 0, …},
  }
},

assignments: {
.."voltage_level_vl1": {
   "voltage_level_components":
    {
    "vl1_di3": { status = "on"},
    "vl1_di4": { status = "on"}
    }
 }
}
```

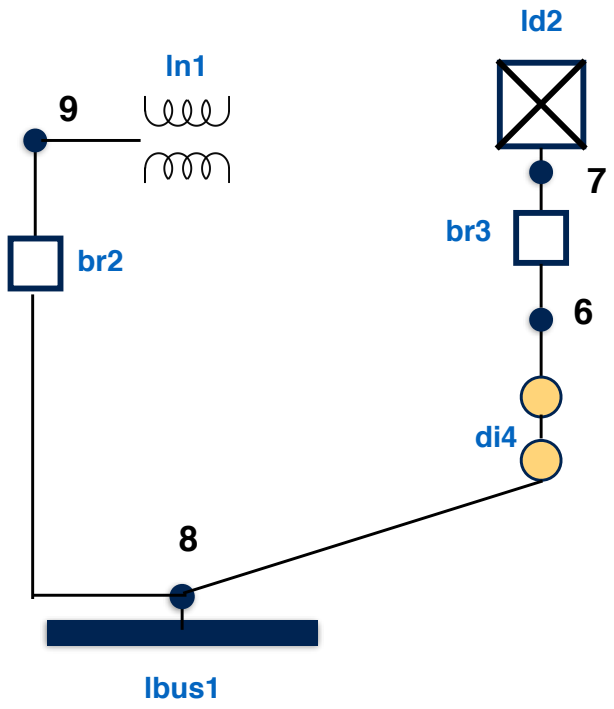
GRG Script: Node-breaker To Bus-breaker




```
"voltage_level_vl1" :
{
  "id" : "vl1",
  "type" : "voltage_level",
  "voltage" : {
    "lower_limit" : 59.0,
    "nominal_value" : 63.0,
    "upper_limit" : 68.0
  },
  "voltage_level_components" :
  {
    “vl1_lbus1” : {“link”: 8, … },
    "vl1_ld2" : {“link”: 7, …},
    "vl1_br2" : {“link_1”: 9, “link_2”: 8, …},
    "vl1_br3" : {“link_1”: 7, “link_2”: 6, …},
    "vl1_di4" : {“link_1”: 6, “link_2”: 8, …},
  }
},

assignments: {
..“voltage_level_vl1”: {
   “voltage_level_components”:
    {
   "vl1_di4": { status = “on"}
    }
 }
}
```

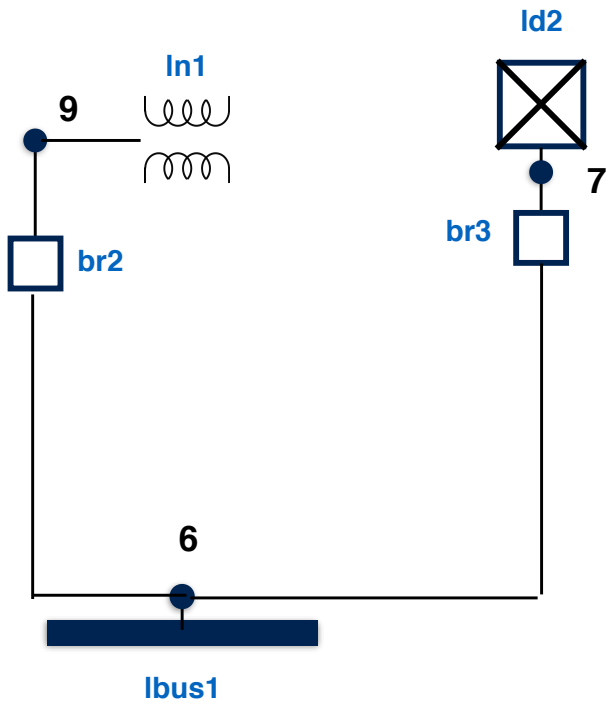
GRG Script: Node-breaker To Bus-breaker




```
"voltage_level_vl1" :
{
  "id" : "vl1",
  "type" : "voltage_level",
  "voltage" : {
    "lower_limit" : 59.0,
    "nominal_value" : 63.0,
    "upper_limit" : 68.0
  },
  "voltage_level_components" :
  {
    “vl1_lbus1” : {“link”: 8, … },
    “vl1_ld2” : {“link”: 7, …},
    “vl1_br2” : {“link_1”: 9, “link_2”: 8, …},
    “vl1_br3” : {“link_1”: 7, “link_2”: 6, …},
    “vl1_di4” : {“link_1”: 6, “link_2”: 8, …},
  }
},

assignments: {
..“voltage_level_vl1”: {
   “voltage_level_components”:
    {
   “vl1_di4”: { status = “on”}
    }
 }
}
```

GRG Script: Node-breaker To Bus-breaker




```
"voltage_level_vl1" :
{
  "id" : "vl1",
  "type" : "voltage_level",
  "voltage" : {
    "lower_limit" : 59.0,
    "nominal_value" : 63.0,
    "upper_limit" : 68.0
  },
  "voltage_level_components" :
  {
    “vl1_lbus1” : {“link”: 6, … },
    "vl1_ld2" : {“link”: 7, …},
    "vl1_br2" : {“link_1”: 9, “link_2”: 6, …},
    "vl1_br3" : {“link_1”: 7, “link_2”: 6, …},
  }
},

assignments: {
..“voltage_level_vl1”: {
   “voltage_level_components”:
    {
    }
 }
}
```

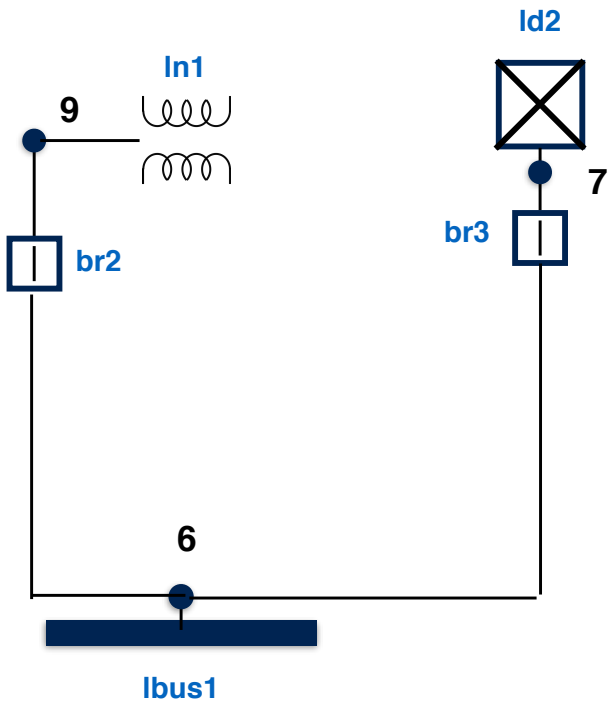
GRG Script: Assign

```
{
   "grid_version": “0.1.0a0",
   "script": {
      "scope": “/…/input_networks/”,
      "network_filename": “demo_net”,
      "steps": [
         {
            "name": “net_0”,
            "type": “load",
            "status": “on”
         },
         {
            "name": “net_1",
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "disconnectors_assignment"
         },
         {
            "name": “net_2”,
            "type": “transform",
            "subtype": “nodebreaker_to_busbreaker",
            "status": “on"
         },
         {
            "name": “net_3”,
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "breakers_assignment"
         },
         {
            "name": “net_4”,
            "type": “transform",
            "subtype": “busbreaker_to_busbranch",
            "status": “on"
         },
         {
            "name": “net_5”,
            "type": “transform",
            “subtype”: “per_unit”,
            “nominal_power”: 100.00,
            "status": “on"
         },
         {
            "name": “net_6”,
            "type": “solve",
            "subtype": “ac_opf”,
            "status": “on”,
            "solver": “ipopt",
            "relaxation": “ACPOL”,
            “starting_points”: {“apply_assignment”:
            “starting_points"}
         },
         {
            “name”: “net_7”,
            "type": “rollback",
            "target": “net_0“,
            "status": “on”
         },
         {
            "type": “output”,
            “subtype”: “network”,
            "status": “on"
            “file_out”: “/…/output_networks/demo_net”
         }
      ]
   }
}
```

GRG Script: Bus-breaker to Bus-branch

```
{
   "grid_version": “0.1.0a0",
   "script": {
      "scope": “/…/input_networks/”,
      "network_filename": “demo_net”,
      "steps": [
         {
            "name": “net_0”,
            "type": “load",
            "status": “on”
         },
         {
            "name": “net_1",
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "disconnectors_assignment"
         },
         {
            "name": “net_2”,
            "type": “transform",
            "subtype": “nodebreaker_to_busbreaker",
            "status": “on"
         },
         {
            "name": “net_3”,
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "breakers_assignment"
         },
         {
            "name": “net_4”,
            "type": “transform",
            "subtype": “busbreaker_to_busbranch",
            "status": “on"
         },
         {
            "name": “net_5”,
            "type": “transform",
            “subtype”: “per_unit”,
            “nominal_power”: 100.00,
            "status": “on"
         },
         {
            "name": “net_6”,
            "type": “solve",
            "subtype": “ac_opf”,
            "status": “on”,
            "solver": “ipopt",
            "relaxation": “ACPOL”,
            “starting_points”: {“apply_assignment”:
            “starting_points"}
         },
         {
            “name”: “net_7”,
            "type": “rollback",
            "target": “net_0“,
            "status": “on”
         },
         {
            "type": “output”,
            “subtype”: “network”,
            "status": “on"
            “file_out”: “/…/output_networks/demo_net”
         }
      ]
   }
}
```

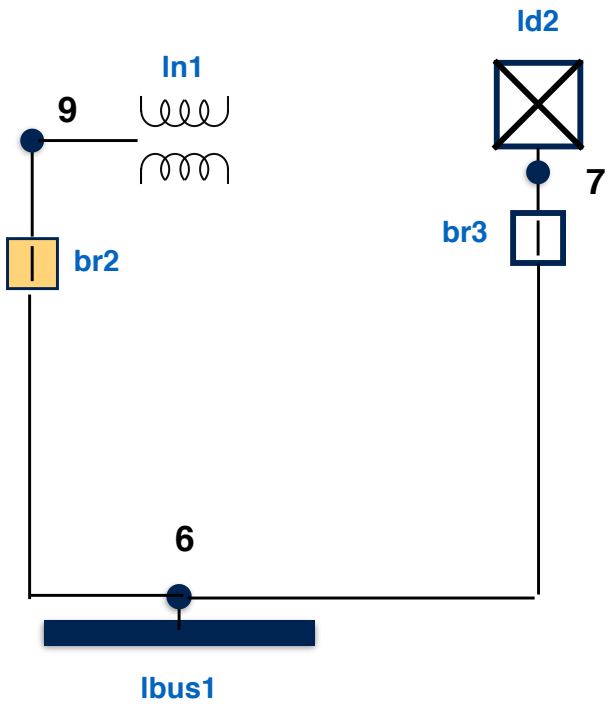
GRG Script: Bus-breaker to Bus-branch




```
"voltage_level_vl1" :
{
  "id" : "vl1",
  "type" : "voltage_level",
  "voltage" : {
    "lower_limit" : 59.0,
    "nominal_value" : 63.0,
    "upper_limit" : 68.0
  },
  "voltage_level_components" :
  {
    “vl1_lbus1” : {“link”: 6, … },
    "vl1_ld2" : {“link”: 7, …},
    "vl1_br2" : {“link_1”: 9, “link_2”: 6, …},
    "vl1_br3" : {“link_1”: 7, “link_2”: 6, …},
  }
},
“ln1”: {“link1”: “9”, “link2”: 3, …}

assignments: {
..“voltage_level_vl1”: {
   “voltage_level_components”:
    {
      “vl1_br2”: {“status”: “on”},
      “vl1_br3”: {“status”: “on”}
    }
 }
}
```

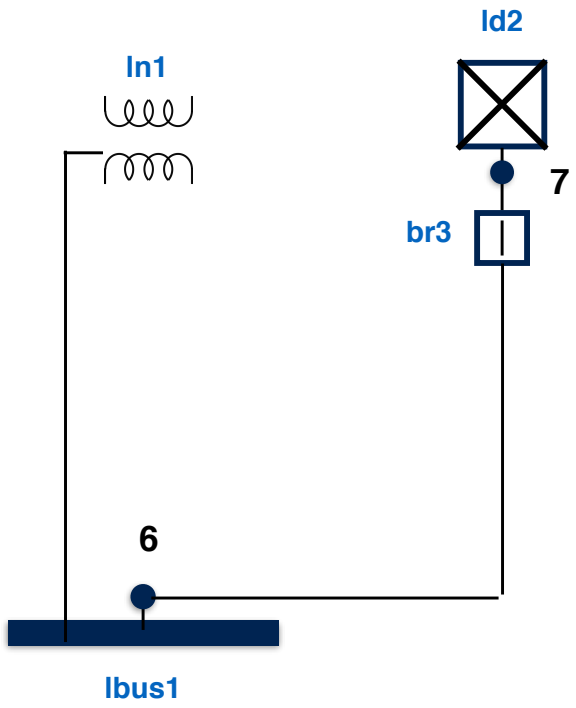
GRG Script: Bus-breaker to Bus-branch




```
"voltage_level_vl1" :
{
  "id" : "vl1",
  "type" : "voltage_level",
  "voltage" : {
    "lower_limit" : 59.0,
    "nominal_value" : 63.0,
    "upper_limit" : 68.0
  },
  "voltage_level_components" :
  {
    “vl1_lbus1” : {“link”: 6, … },
    "vl1_ld2" : {“link”: 7, …},
    "vl1_br2" : {“link_1”: 9, “link_2”: 6, …},
    "vl1_br3" : {“link_1”: 7, “link_2”: 6, …},
  }
},
“ln1”: {“link1”: “9”, “link2”: 3, …}

assignments: {
..“voltage_level_vl1”: {
   “voltage_level_components”:
    {
      “vl1_br2”: {“status”: “on”},
      “vl1_br3”: {“status”: “on”}
    }
 }
}
```

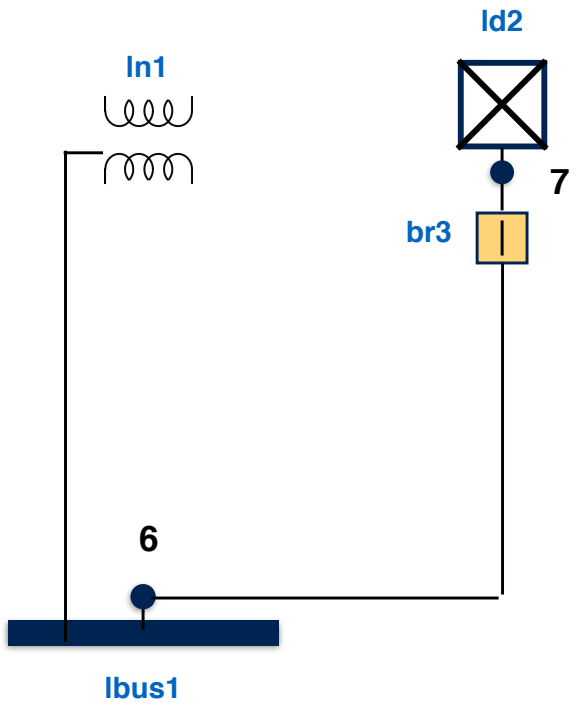
GRG Script: Bus-breaker to Bus-branch




```
"voltage_level_vl1" :
{
  "id" : "vl1",
  "type" : "voltage_level",
  "voltage" : {
    "lower_limit" : 59.0,
    "nominal_value" : 63.0,
    "upper_limit" : 68.0
  },
  "voltage_level_components" :
  {
    “vl1_lbus1” : {“link”: 6, … },
    "vl1_ld2" : {“link”: 7, …},
    "vl1_br3" : {“link_1”: 7, “link_2”: 6, …},
  }
},
“ln1”: {“link1”: “lbus1”, “link2”: 3, …}

assignments: {
..“voltage_level_vl1”: {
   “voltage_level_components”:
    {
      “vl1_br3”: {“status”: “on”}
    }
 }
}
```

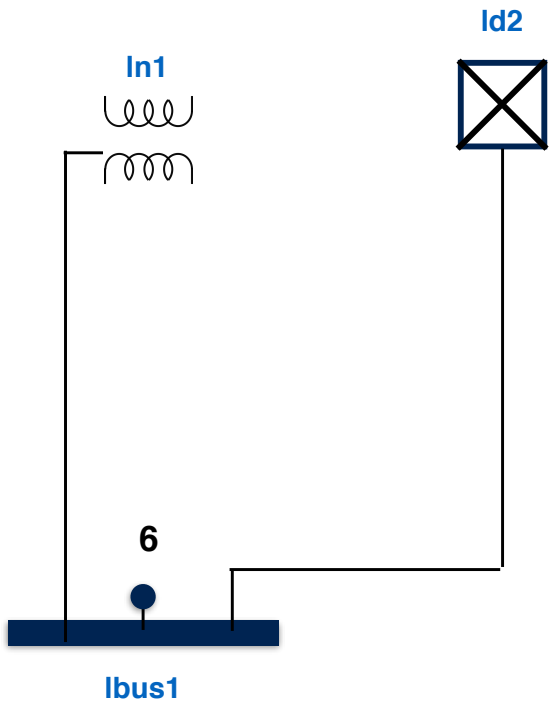
GRG Script: Bus-breaker to Bus-branch




```
"voltage_level_vl1" :
{
  "id" : "vl1",
  "type" : "voltage_level",
  "voltage" : {
    "lower_limit" : 59.0,
    "nominal_value" : 63.0,
    "upper_limit" : 68.0
  },
  "voltage_level_components" :
  {
    “vl1_lbus1” : {“link”: 6, … },
    "vl1_ld2" : {“link”: 7, …},
    "vl1_br3" : {“link_1”: 7, “link_2”: 6, …},
  }
},
“ln1”: {“link1”: “lbus1”, “link2”: 3, …}

assignments: {
..“voltage_level_vl1”: {
   “voltage_level_components”:
    {
      “vl1_br3”: {“status”: “on”}
    }
 }
}
```

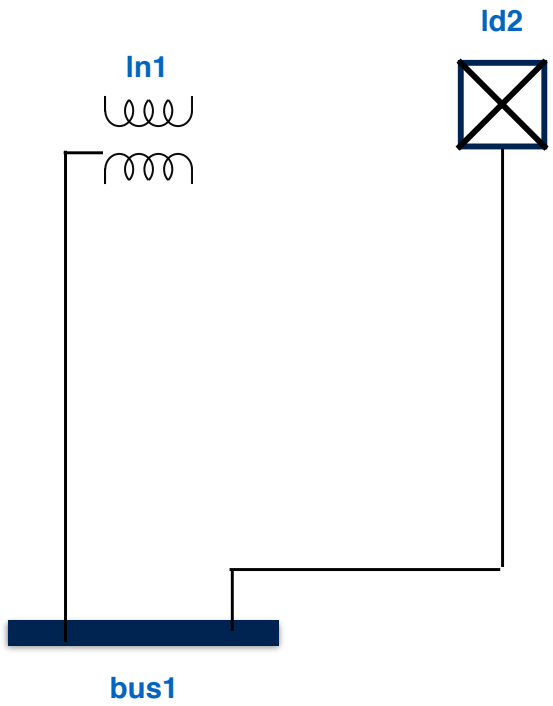
GRG Script: Bus-breaker to Bus-branch




```
"voltage_level_vl1" :
{
  "id" : "vl1",
  "type" : "voltage_level",
  "voltage" : {
    "lower_limit" : 59.0,
    "nominal_value" : 63.0,
    "upper_limit" : 68.0
  },
  "voltage_level_components" :
  {
    “vl1_lbus1” : {“link”: 6, … },
    "vl1_ld2" : {“link”: “lbus1”, …},
  }
},
“ln1”: {“link1”: “lbus1”, “link2”: 3, …}

assignments: {
..“voltage_level_vl1”: {
   “voltage_level_components”:
    {
    }
 }
}
```

GRG Script: Bus-breaker to Bus-branch




```
"voltage_level_vl1" :
{
  "id" : "vl1",
  "type" : "voltage_level",
  "voltage" : {
    "lower_limit" : 59.0,
    "nominal_value" : 63.0,
    "upper_limit" : 68.0
  },
  "voltage_level_components" :
  {
    “vl1_lbus1” : {… },
    "vl1_ld2" : {“link”: “bus1”, …},
  }
},
“ln1”: {“link1”: “bus1”, “link2”: 3, …}

assignments: {
..“voltage_level_vl1”: {
   “voltage_level_components”:
    {
    }
 }
}
```

GRG Script: Per-unit transform

```
{
   "grid_version": “0.1.0a0",
   "script": {
      "scope": “/…/input_networks/”,
      "network_filename": “demo_net”,
      "steps": [
         {
            "name": “net_0”,
            "type": “load",
            "status": “on”
         },
         {
            "name": “net_1",
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "disconnectors_assignment"
         },
         {
            "name": “net_2”,
            "type": “transform",
            "subtype": “nodebreaker_to_busbreaker",
            "status": “on"
         },
         {
            "name": “net_3",
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "breakers_assignment"
         },
         {
            "name": “net_4”,
            "type": “transform",
            "subtype": “busbreaker_to_busbranch",
            "status": “on"
         },
         {
            "name": “net_5”,
            "type": “transform",
            “subtype”: “per_unit”,
            “nominal_power”: 100.00,
            "status": “on"
         },
         {
            "name": “net_6”,
            "type": “solve",
            "subtype": “ac_opf”,
            "status": “on”,
            "solver": “ipopt",
            "relaxation": “ACPOL”,
            “starting_points”: {“apply_assignment”:
            “starting_points"}
         },
         {
            “name”: “net_7”,
            "type": “rollback",
            "target": “net_0“,
            "status": “on”
         },
         {
            "type": “output”,
            “subtype”: “network”,
            "status": “on"
            “file_out”: “/…/output_networks/demo_net”
         }
      ]
   }
}
```

GRG Script: Per-unit transform

input

```
// Component
{
   "subtype" : "solar",
   "type" : "generator",
   "voltage_regulation" : "off"
   "link" : 5,
   "output" : {
  "active" : {
   "var" : {"lb" : 0, "ub" : 20.7 }
  },
  "reactive" : {
   "var" : {"lb" : 0, "ub" : 0.0 }
       }
   },
  "PQ_curve" : [
  { "active" : 0,
  "reactive" : {"lower_limit" : 0, "upper_limit" : 0}
  },
   {"active" : 20.699999999999999,
  "reactive" : {"lower_limit" : 0, "upper_limit" : 0}
   }
   ]
}

// Assignments
{ }

// Mappings
target_points: {
   "output" : {
  "active" : 0, "reactive" : 0, "voltage" : 65.744705}
   },
starting_points: {
   "output" : {
  "active" : 0, "reactive" : 0}
   }
}
```

output

```
// Component
{ . . . }

// Assignments
{
   "output" : {
  "active" : {
   "var" : {"lb" : 0, "ub" : 0.207}
  },
  "reactive" : {
   "var" : {"lb" : 0, "ub" : 0.0 }
       }
   },
  "PQ_curve" : [
  { "active" : 0,
  "reactive" : {"lower_limit" : 0, "upper_limit" : 0}
  },
   {"active" : 0.207,
  "reactive" : {"lower_limit" : 0, "upper_limit" : 0}
   }
   ]
}

// Mappings
target_points: {
   "output" : {
  "active" : 0, "reactive" : 0, "voltage" : 1.0435667}
   },
starting_points: {
   "output" : {
  "active" : 0, "reactive" : 0}
   }
}
```

GRG Script: Solve

```
{
   "grid_version": “0.1.0a0",
   "script": {
      "scope": “/…/input_networks/”,
      "network_filename": “demo_net”,
      "steps": [
         {
            "name": “net_0”,
            "type": “load",
            "status": “on”
         },
         {
            "name": “net_1",
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "disconnectors_assignment"
         },
         {
            "name": “net_2”,
            "type": “transform",
            "subtype": “nodebreaker_to_busbreaker",
            "status": “on"
         },
         {
            "name": “net_3",
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "breakers_assignment"
         },
         {
            "name": “net_4”,
            "type": “transform",
            "subtype": “busbreaker_to_busbranch",
            "status": “on"
         },
         {
            "name": “net_5”,
            "type": “transform",
            “subtype”: “per_unit”,
            “nominal_power”: 100.00,
            "status": “on"
         },
         {
            "name": “net_6”,
            "type": “solve",
            "subtype": “ac_opf”,
            "status": “on”,
            "solver": “ipopt",
            "relaxation": “ACPOL”,
            “starting_points”: {“apply_assignment”:
            “starting_points"}
         },
         {
            “name”: “net_7”,
            "type": “rollback",
            "target": “net_0“,
            "status": “on”
         },
         {
            "type": “output”,
            “subtype”: “network”,
            "status": “on"
            “file_out”: “/…/output_networks/demo_net”
         }
      ]
   }
}
```

GRG Script: Rollback

```
{
   "grid_version": “0.1.0a0",
   "script": {
      "scope": “/…/input_networks/”,
      "network_filename": “demo_net”,
      "steps": [
         {
            "name": “net_0”,
            "type": “load",
            "status": “on”
         },
         {
            "name": “net_1",
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "disconnectors_assignment"
         },
         {
            "name": “net_2”,
            "type": “transform",
            "subtype": “nodebreaker_to_busbreaker",
            "status": “on"
         },
         {
            "name": “net_3",
            "type": “transform",
            "subtype": “assign",
            "status": “on",
            "apply_assignment": "breakers_assignment"
         },
         {
            "name": “net_4”,
            "type": “transform",
            "subtype": “busbreaker_to_busbranch",
            "status": “on"
         },
         {
            "name": “net_5”,
            "type": “transform",
            “subtype”: “per_unit”,
            “nominal_power”: 100.00,
            "status": “on"
         },
         {
            "name": “net_6”,
            "type": “solve",
            "subtype": “ac_opf”,
            "status": “on”,
            "solver": “ipopt",
            "relaxation": “ACPOL”,
            “starting_points”: {“apply_assignment”:
            “starting_points"}
         },
         {
            “name”: “net_7”,
            "type": “rollback",
            "target": “net_0“,
            "status": “on”
         },
         {
            "type": “output”,
            “subtype”: “network”,
            "status": “on"
            “file_out”: “/…/output_networks/demo_net”
         }
      ]
   }
}
```

Overview

- The team
- Project objectives
- Technical approach
- ***Accomplishments***
  - test cases
- Goals
- Tech-to-Market
- Collaborations

Four Networks

Network Components Summary

		Marseille	France EHV	Lyon	France
Substations		74	1174	2376	4749
Voltage Levels		100	1523	2951	5742
Busbars		196	3305	5842	11644
Lines		112	1968	3775	7662
Transformers		43	558	937	1688
Generators		53	692	1202	2197
	Hydro	22	433	645	792
	Nuclear	0	59	59	59
	Solar	31	90	269	643
	Thermal	0	65	74	112
	Wind	0	43	152	587
Loads		142	2184	3750	6993
Shunt		15	196	214	381
Switches		1396	24818	41050	76461
	Breakers	553	8733	15003	28889
	Disconnectors	840	15832	25380	46404

Validation

- Validation of every component model
  - unit tests from load flow results
  - accuracy around 10^{-3}
  - validation is not performed in the per-unit system
    - load flow results are in nominal values
    - may be an interesting thing to add

Metrics

- Two threads
  - defining a set of metrics that can capture OPF difficulties
  - comparing various test cases with respect to these metrics
- Defining the metrics
  - work in progress
  - e.g., distances of loads from generators
- Implementing and analyzing the metrics
  - work in progress
  - e.g., understanding why a network exhibits some peculiar values
  - e.g., linking the metrics to computational difficulties

Four Networks

Network Components Summary

		Marseille	France EHV	Lyon	France
Substations		74	1174	2376	4749
Voltage Levels		100	1523	2951	5742
Busbars		196	3305	5842	11644
Lines		112	1968	3775	7662
Transformers		43	558	937	1688
Generators		53	692	1202	2197
	Hydro	22	433	645	792
	Nuclear	0	59	59	59
	Solar	31	90	269	643
	Thermal	0	65	74	112
	Wind	0	43	152	587
Loads		142	2184	3750	6993
Shunt		15	196	214	381
Switches		1396	24818	41050	76461
	Breakers	553	8733	15003	28889
	Disconnectors	840	15832	25380	46404

Metrics

Network Parameters-1

		Marseille	France EHV	Lyon	France
Degree	avg	1.55	1.65	1.60	1.64
	min	1	1	1	1
	max	12	20	20	20
	sd	1.85	2.28	2.09	2.07
	median	1	1	1	1
Load-Gen distance	avg	2.93	3.56	3.19	2.92
	min	2	1	1	1
	max	8	11	11	11
	sd	1.97	1.93	1.83	1.76
	median	2	3	3	3
Simple Cycles		92	1429	2423	4886

Metrics

Network Components Summary

	edin_29	ieee_30	pegase_89	ieee_118	ieee_300
Substations	-	-	-	-	-
Voltage Levels	-	-	-	-	-
Busbars	29	30	89	118	300
Lines	95	39	175	177	304
Transformers	4	4	35	9	107
Generators	66	2	12	19	57
Synchronous Condensers	0	4	0	35	12
Loads	29	21	35	99	201
Shunt	0	2	44	14	29

Metrics

Network Parameters

		edin_29	ieee_30	pegase_89	ieee_118	ieee_300
Degree	avg	6.82	2.37	4.72	3.15	2.74
	min	4	1	1	1	1
	max	12	7	15	12	12
	sd	2.18	1.39	4.05	1.80	1.56
	median	6	2	3	2	3
Load-Gen distance	avg	1.17	4.28	4.71	2.70	3.25
	min	1	1	2	1	1
	max	2	7	8	5	8
	sd	0.38	1.69	1.48	1.47	1.66
	median	1	4	5	3	3
Simple Cycles		28	21	135	80	163

Metrics (Marseille Sous Realtor)

Network Components Summary

Substations	285
Voltage Levels	365
Busbar	4781
Lines	452
Transformers	122
Synchronous Condensers	0
Loads	594
Shunt	42

Generator Summary

Total	177
Solar	76
Wind	5
Thermal	15
Nuclear/Hydro	81
Other	0

Switch Summary

Total	9223
Breaker	2142
Disconnector	7057

Metrics (Marseille Sous Realtor)

Line Impedance

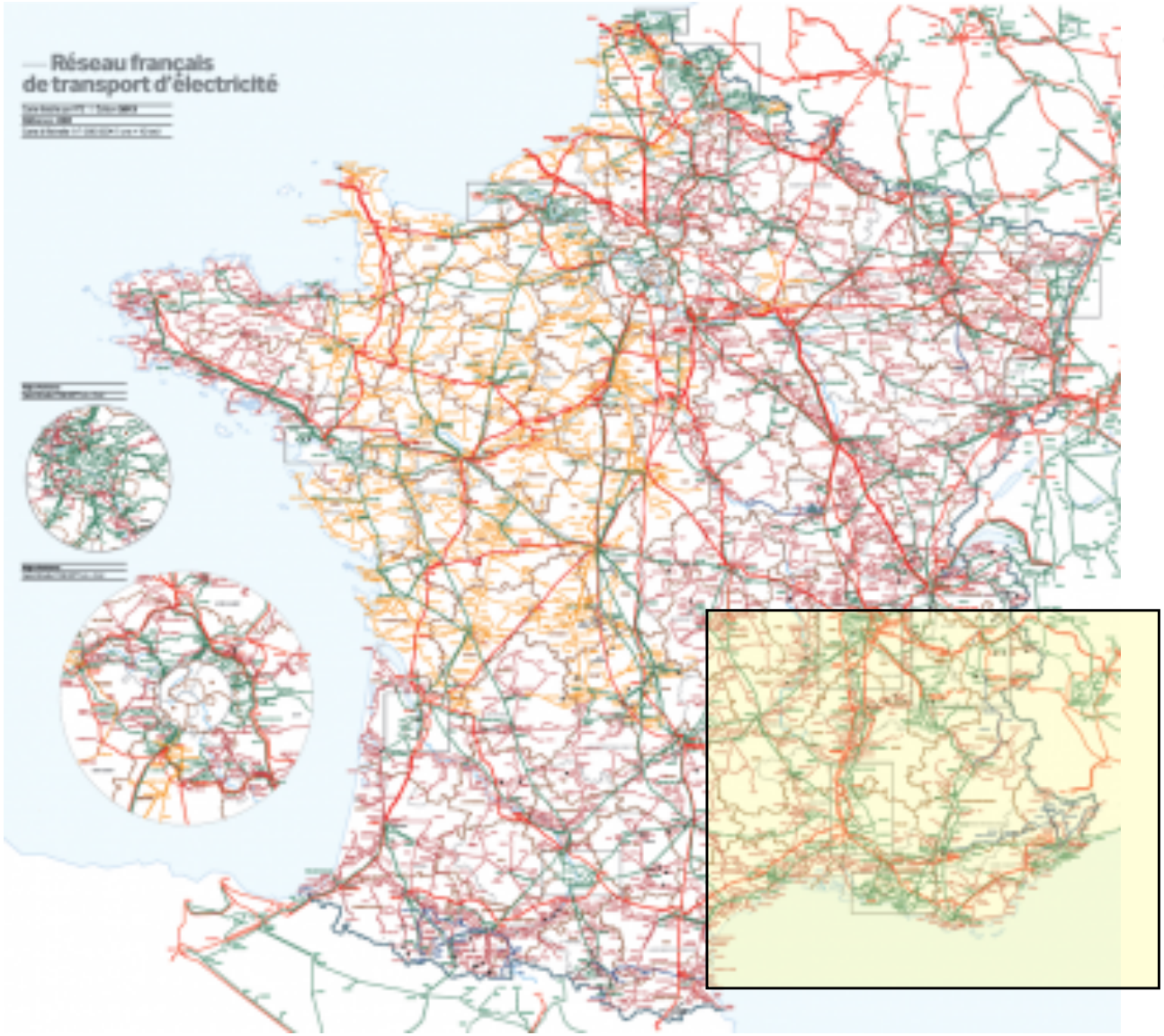
Impedance (Ohms)		
Reactance	avg	3.12
	min	0.00
	max	31.93
	sd	3.57
	median	2.23
Resistance	avg	0.89
	min	0.00
	max	6.98
	sd	0.94
	median	0.64

Network distances

Load to Generator		
Distance	avg	3.17
	min	1
	max	9
	sd	1.96
	median	3

Metrics (Marseille Sous Realtor)

Metrics (Marseille Sous Realtor)



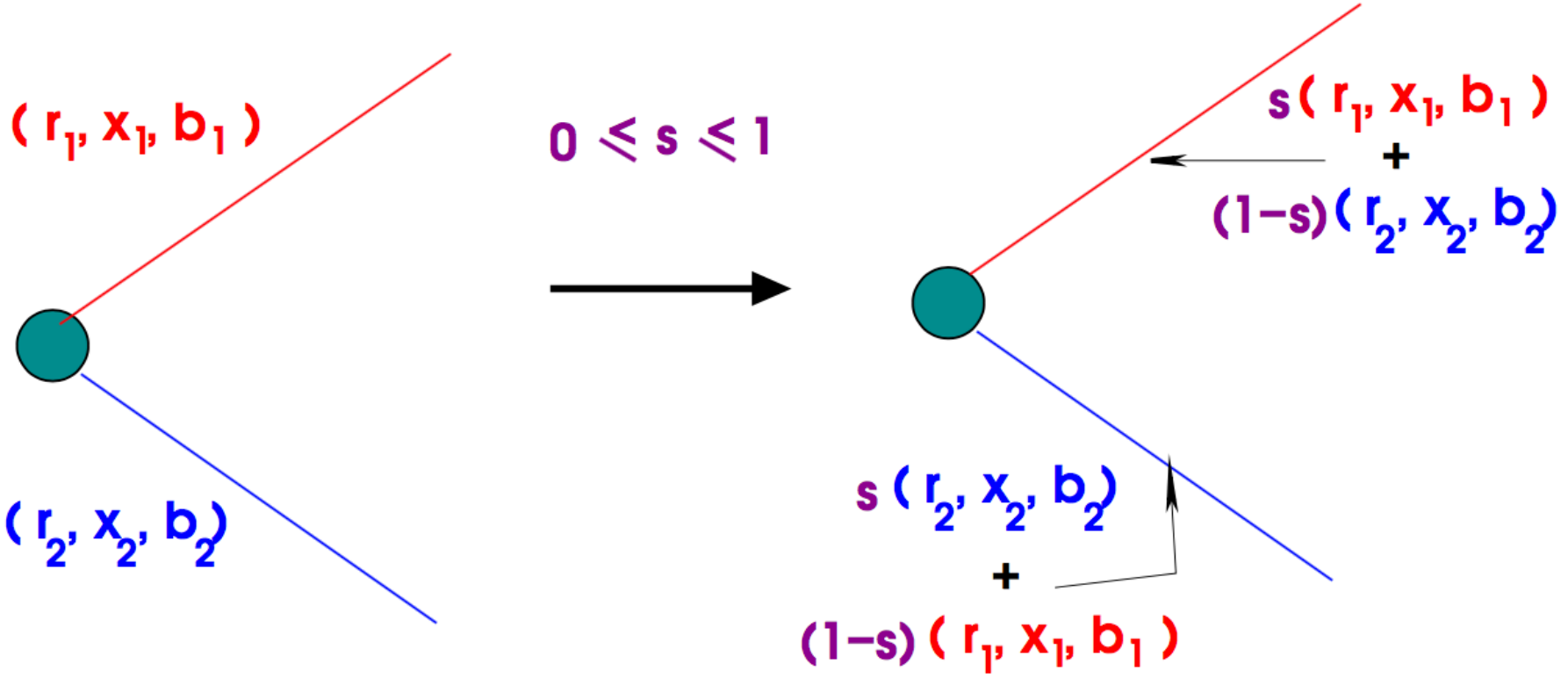
Overview

- The team
- Project objectives
- Technical approach
- Accomplishments
- ***Goals***
- Tech-to-Market
- Collaborations

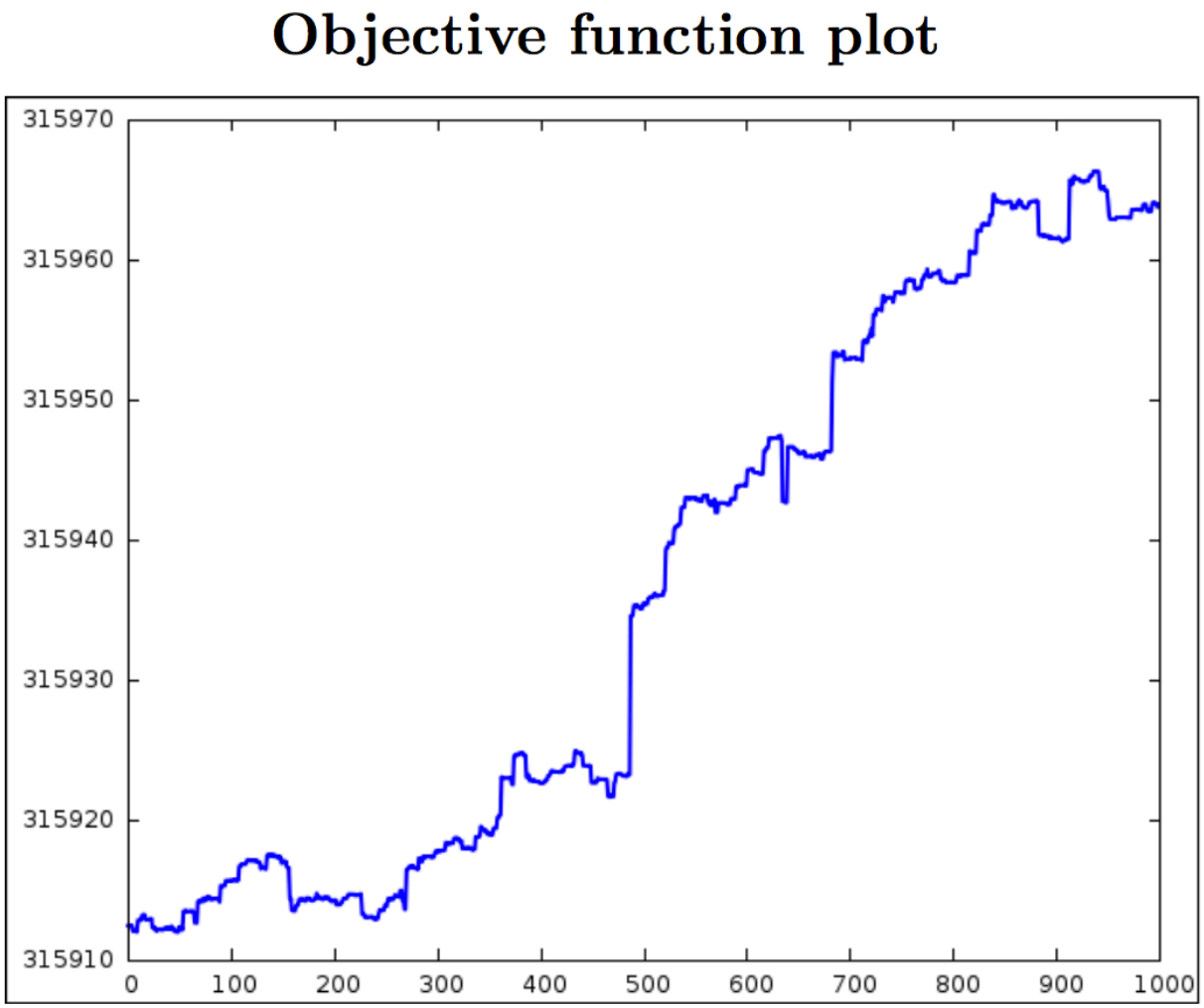
Synthetic Test Cases

- Specification of a static methodology
  - keep the topology
  - vary the line/load/generation parameters
- Interesting observations
  - not hard to generate instances that are infeasible
  - but almost all modifications do not create infeasibilities
  - objective function does not vary too much
- Next step
  - looking closely at the transition phase

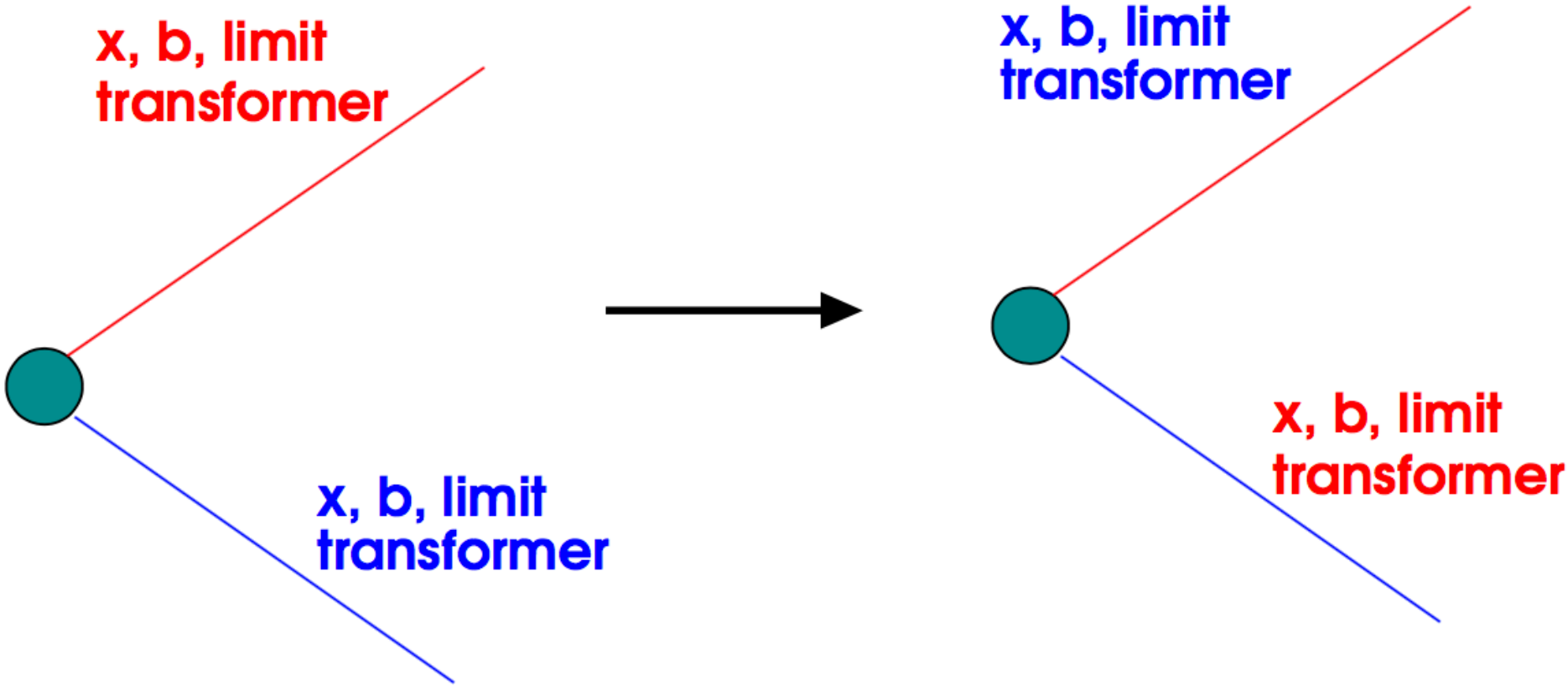
Impedance blending



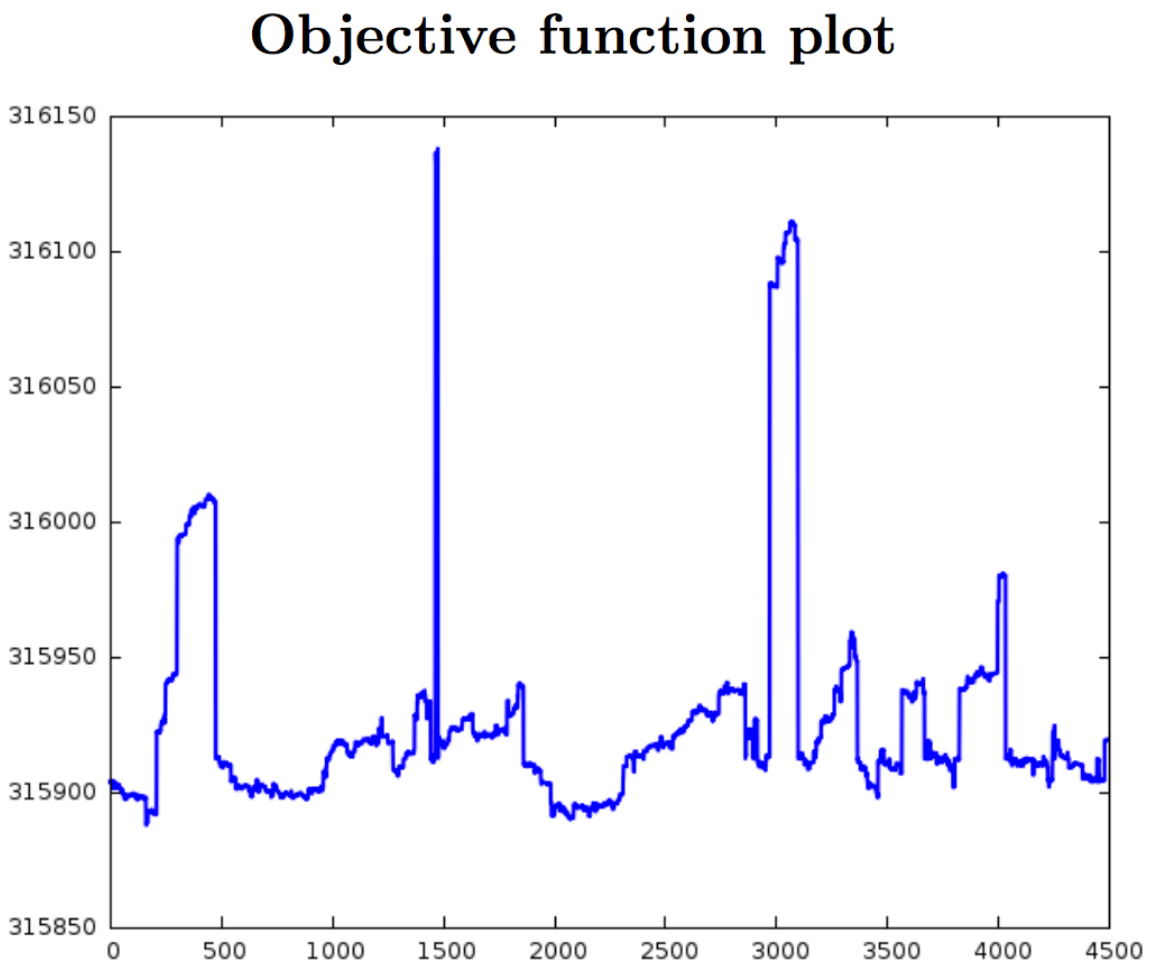
Synthetic data



Line parameters



Line Parameters



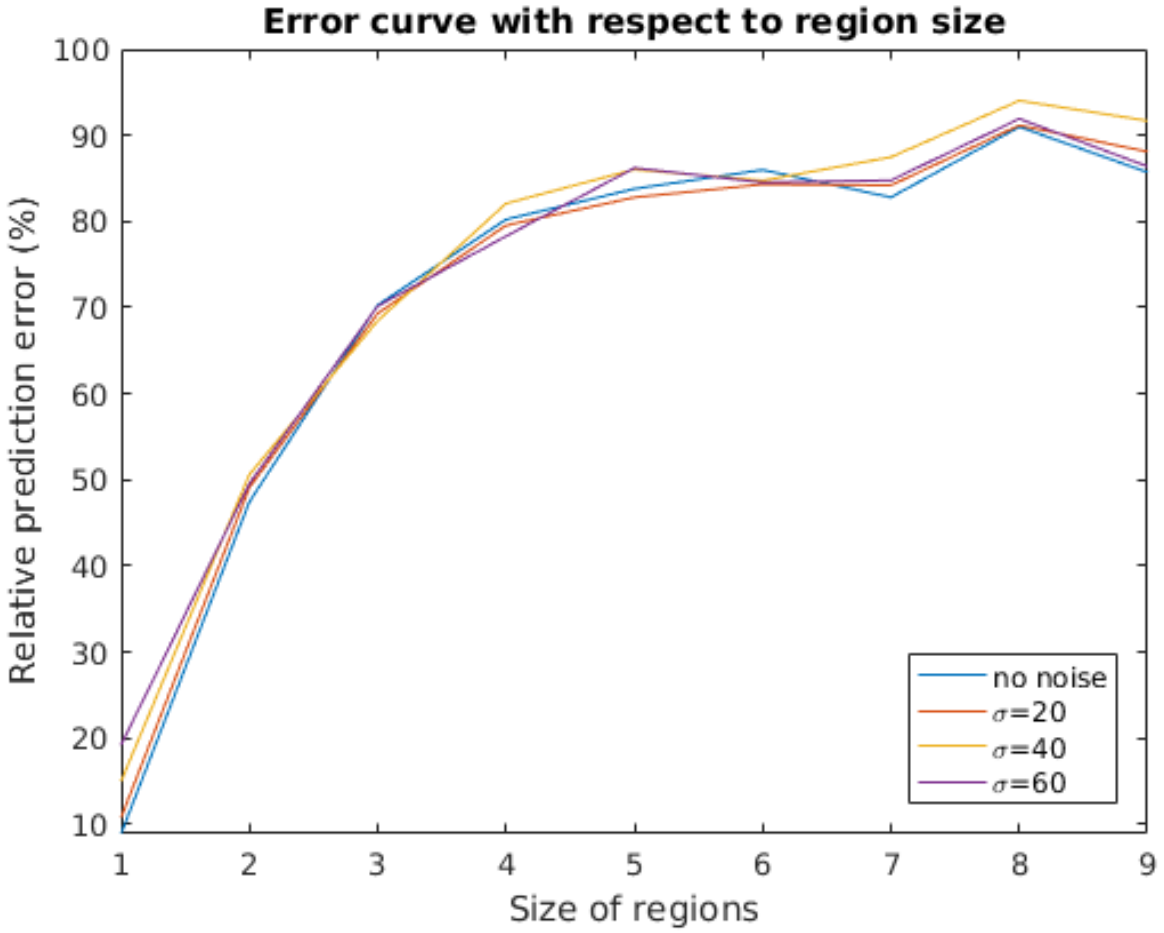
Disaggregation

- Develop models and algorithms to disaggregate loads and generations by type and locations
  - Based on available data at RTE
- Example data include
  - Transmission network model
  - Coarse timescale of aggregate generations and loads
  - Fine timescale data at a few locations

Disaggregation

- Given:
  - network data (topology, impedances, switch status)
  - aggregate generation and load observations (time series)
  - AMI or probabilistic information on part of network
- Characterize:
  - underlying states (disaggregated data, “hidden” states) that are most consistent with observations

Disaggregation



$$\text{relative error} := \frac{1}{N}\sum_{i=1}^{N}\frac{\left|\hat{s}_i^* - s_i\right|}{\left|s_i\right|}$$

Results

- Naive algorithm performs poorly
- Error due to topology effect dominates measurement noise
- Need to develop qualitative theory first

Overview

- The team
- Project objectives
- Technical approach
- Accomplishments
- Goals
- ***Tech-to-Market***
- Collaborations

Potential Impact

- High-fidelity test cases
  - network + modeling
- New format for the necessary fidelity
  - extensible, flexible, light-weight
- Tools for manipulating network data
  - transformation, conversion, roll-back

Overview

- The team
- Project objectives
- Technical approach
- Accomplishments
- Goals
- Tech-to-Market
- ***Collaborations***

Opportunities for GRID DATA Collaborations

- Converting other test cases in GRG
  - very interested in extending the format
  - additional components (e.g., HVDC lines)
  - students and post-doc exchanges

- Industrial collaborations
  - modeling contingencies
  - topology optimization

- Metrics
  - testing metrics and understanding test cases
  - very interested in getting interesting metrics
- Synthetic test cases
  - complementary approaches

Conclusions

- High fidelity modeling
  - network, models
  - targeted to the needs of industry
- Novel flexible and extensive format
  - JSON + suite of tools
- Real test cases
  - unit validation at this stage
- Synthetic test cases
  - promising experiments
- Disaggregation and obfuscation
  - more work to be done