# SynchroGrid

Achieving Simplicity in System Protection

SynchroGrid

Achieving Simplicity in System Protection

# Auto-tuned Solution to Wide Area Coordination Issues of Distance and Directional Time Overcurrent Relay Settings


NATHAN THOMAS, Ph.D.
Software & Automation

LUKE HANKINS, P.E.
Protection Engineer

JOE PEREZ, P.E.
Founder – Protection Engineer



**Presented at the 74th Annual Texas A&M Relay Conference**

# Presentation Outline

- Challenges in Wide Area Coordination
- Related Work and Our Previous Work
- Overview of Approach
- New Capabilities
- Experimental Results
- Conclusion and Future Work

# Wide Area Coordination

*"Wide-area coordination (WAC) analysis is the evaluation of protective device selectivity and sensitivity at a system level (multiple layers of adjacent terminals) with a goal of improving system reliability" [Barman 2014].*

- Wide-area coordination is a labor intensive, difficult process.
- Automated tools to verify coordination exist, but less work has been done to automate development of coordinated settings to avoid (or fix) reported violations.
- **PRC-027-1** is here! WAC must be performed with increasing frequency.

# Wide Area Coordination

*Specific Challenges*

- Directional Time Overcurrent Relays – three parameters (pickup, time dial, curve type) collectively define complex response of the element. Very difficult in tightly coupled systems. *[Thomas et al, 2019].*

- Multi-element coordination – not just OC/OC and DS/DS pairs, but OC/DS, etc. Former may be too restrictive (or hide violations).

- How to handle unresolvable miscoordinations.

# Our Approach

Use an ***auto-tuner*** which can quickly create coordinated relay settings based on guidance from an engineer.

Leverage our previous work for the automation workflow.

- Input collection (e.g., backup relay pairs, fault currents) from short circuit program [Boecker, Corpuz, Perez, Hankins 2017].
- Automated coordination verification [C. Thomas, Perez, Hankins, Tribur 2018].
- Directional Time Overcurrent Relay auto-tuning (pickups and time dials) [N. Thomas, Hankins, Perez 2019].

In this work, we address more real-world concerns to move towards a viable, general-purpose tool.

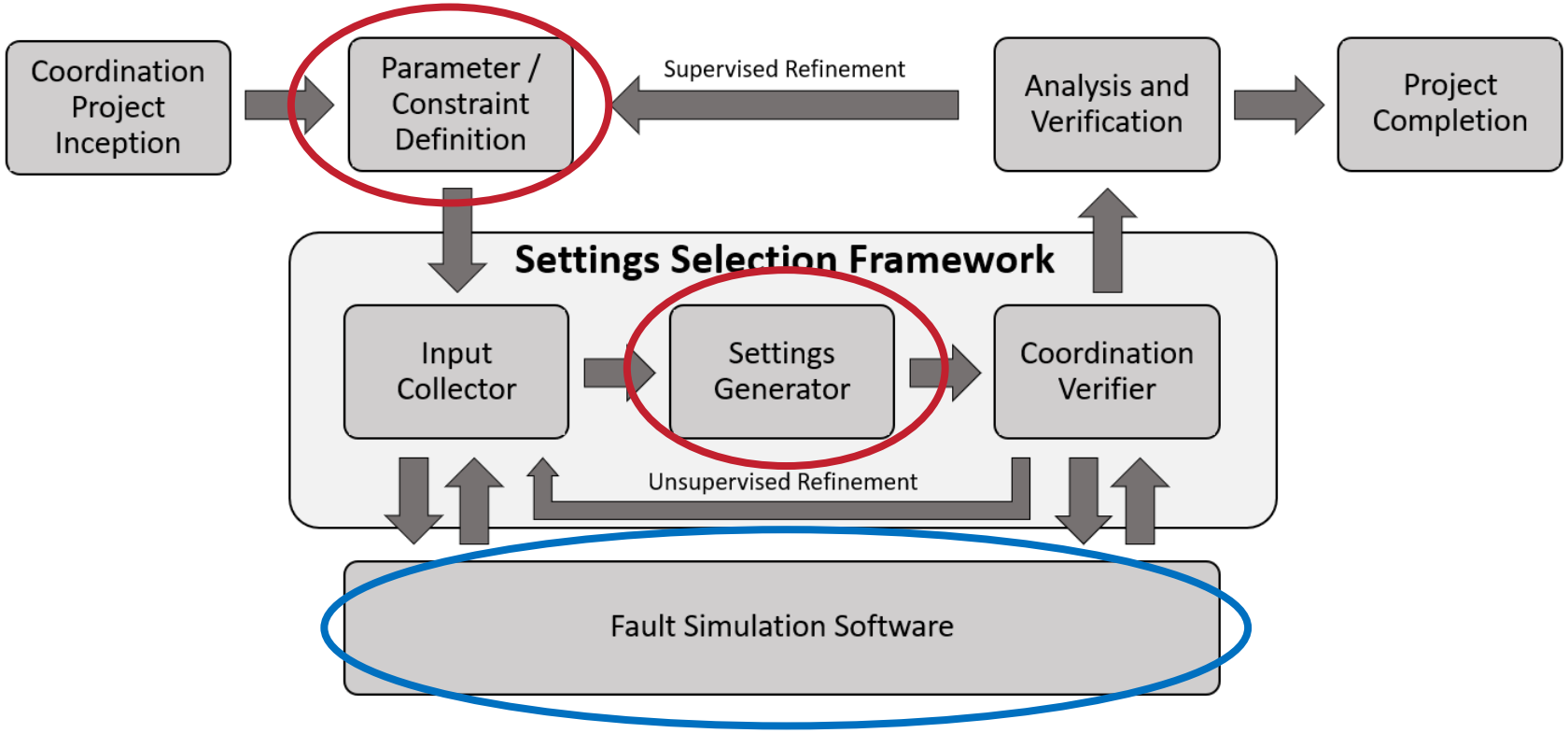
- Support for more timing constraints, OC Curve selection, OC / DS coordination, contingency support, and selective coordination relaxation when full coordination is not possible.

# Previous Work in Auto-tuning

Academic literature is rich in ideas and tracks trends in computer science, machine learning, and artificial intelligence.

- Two surveys
  *[Moirangthem 2011] & [Birla 2005]*
- Optimization Theory
  *[Urdaneta 1988], [Birla 2006], [Damchi 2018]*
- Many other approaches:
  - Genetic algorithms [Ravazi 2008]
  - Reinforcement learning [Kilickiran 2018]
  - Graph theory [Madani 1998]

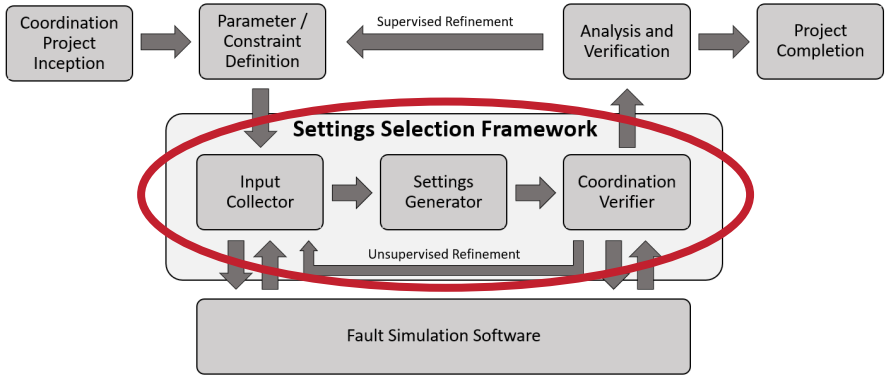
# Auto-Tuning Coordination Workflow




- Primary areas of new work are richer problem *definition* and extension of the *settings generator* to support definition additions.

- Tighter integration with *fault simulation software*. The move from scripting to the linkable library means the settings generator can collaborate better to verify partial solutions during the solve phase.

# Workflow Implementation Overview




- *Input Collector* based on [Boecker, Corpuz, Perez, Hankins 2017].
- *Settings Generator* solver based on optimization theory.
  - Initial formulation is a non linear problem as in [Urdaneta 1988].
  - Transformed to Mixed Integer Linear Programming (MILP) formulation of problem [Damchi 2018].
  - Expressed as large collection of constraints to satisfy such as:
    $OpTime(BackupRlyCfg_i, Cont_j) - OpTime(PrimaryRlyCfg_k, Cont_j) > 1/3$ *second*
  - Modern C++20 codebase. Functional, composable design.
- *Coordination Verifier* based on [Thomas, Perez, Hankins, Tribur 2018].

# Experimental Setup

- ASPEN OneLiner V14.8, latest OlxAPI programming interface.
- Quad core Intel Core i7-8565U and containing 16GB of RAM (laptop).
- All relays tuned are ground relays (can handle phase too).
- Single line to ground faults used in fault studies.
- Standard fault regime includes close-in, close-in end-opened, line-end, remote bus, and intermediate faults at every 10% of the line.
- Coordination Time Interval (CTI) >= 0.33s / 20 cycles.
- Default is for autotuner to prefer solutions minimizing aggregate response time to close-in EO faults.
- Default pickup range is [1..16] (secondary) and time dials [0.5, 16].

# Expanding Constraint Support

- In prior work, CTI enforcement, required fault response, pickup / time dial ranges were only *constraint* placed on the solver. *Insufficient to implement common protection standards.*
- Have integrated more customizable constraint support with small symbolic language and math operators as in [Corpuz 2017]. Expressions are automatically translated linear optimization formulation in the *Settings Generator* and passed to the settings solver.
- For this paper, explored lower and upper response time constraints faults (e.g., line-end faults).
- Example:

$$PrimaryRelayTocTime_{LE_Fault} >= 0.5$$

- Text file based for now, graphical user interface would be ideal.

# Expanding Constraint Support

## Experimental Setup

- 9-bus / 18-relay system based on ASPEN OneLiner example.
- Baseline 1st experiment with no new additional constraints.
- Fixed U3 Curve.
- Converged to optimal result in 0.4 seconds.

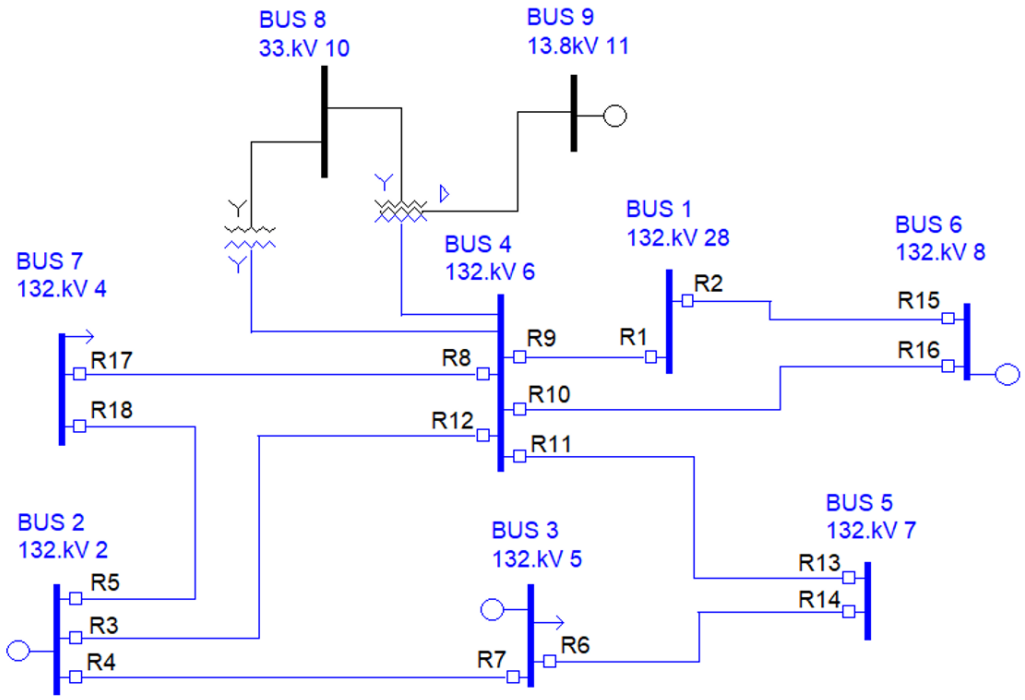



TABLE I
CALCULATED SETTINGS AND RESPONSE FOR 9-BUS SYSTEM (BASELINE)

| Relay | $I_{set}$ | $TMS$ | $T_{LE}$ (s) |
|---|---|---|---|
| R1 | 7.8169 | 2.6402 | 0.3497 |
| R2 | 6.7823 | 2.0038 | 0.4038 |
| R3 | 15.0000 | 0.7609 | 0.4013 |
| R4 | 15.0000 | 1.0987 | 0.5502 |
| R5 | 15.0000 | 1.4380 | 0.3416 |
| R6 | 15.0000 | 1.6069 | 0.5482 |
| R7 | 14.2632 | 0.5099 | 0.3266 |
| R8 | 9.8862 | 0.5000 | 0.6574 |
| R9 | 15.0000 | 2.7469 | 0.4403 |
| R10 | 15.0000 | 1.1459 | 0.2087 |
| R11 | 13.5263 | 0.5000 | 0.1865 |
| R12 | 15.0000 | 1.9859 | 0.5056 |
| R13 | 15.0000 | 1.4023 | 0.3756 |
| R14 | 15.0000 | 0.5000 | 0.3825 |
| R15 | 15.0000 | 0.5000 | 0.4620 |
| R16 | 12.0526 | 0.5000 | 0.5277 |
| R17 | 15.0000 | 1.0837 | 0.6719 |
| R18 | 12.7895 | 0.5000 | 0.4463 |

# Expanding Constraint Support

## Experimental Setup

- 9-bus / 18-relay system based on ASPEN OneLiner example.
- 2nd experiment, *allowed line-end response: [0.5..1.25]sec.*
- Fixed U3 Curve.
- Surprisingly, optimal result achieved in 0.2 seconds *(2x faster)*.
- Branch & bound algorithm aggressively *prunes* solution space.
- *Takeaway* - formulate problem as realistically bound as possible, worry less about making it “too hard”.

TABLE II
CALCULATED SETTINGS AND RESPONSE FOR 9-BUS SYSTEM (LINE END CONSTRAINED)

| Relay | $I_{set}$ | $TMS$ | $T_{LE}$ (s) |
|---|---|---|---|
| R1 | 7.8169 | 3.7752 | 0.5000 |
| R2 | 6.7823 | 2.4815 | 0.5000 |
| R3 | 15.0000 | 0.9609 | 0.5068 |
| R4 | 15.0000 | 1.2026 | 0.6023 |
| R5 | 15.0000 | 2.1051 | 0.5000 |
| R6 | 15.0000 | 1.8178 | 0.6201 |
| R7 | 15.0000 | 0.7068 | 0.5000 |
| R8 | 11.0710 | 0.5000 | 0.8784 |
| R9 | 15.0000 | 3.1193 | 0.5000 |
| R10 | 15.0000 | 2.7453 | 0.5000 |
| R11 | 15.0000 | 1.1308 | 0.5000 |
| R12 | 15.0000 | 2.3653 | 0.6021 |
| R13 | 15.0000 | 1.8665 | 0.5000 |
| R14 | 15.0000 | 0.6536 | 0.5000 |
| R15 | 15.0000 | 0.5411 | 0.5000 |
| R16 | 13.5263 | 0.5157 | 0.7154 |
| R17 | 15.0000 | 1.1888 | 0.7371 |
| R18 | 13.5263 | 0.5000 | 0.5047 |

# Expanding Tunable Parameter Support

## Overcurrent curve parameter selection

- Changing the curve parameter provides more flexibility to resolve difficult coordination situations.
- Expressed in our framework as additional per relay parameter in constraint formulation.
- Support for IEEE and IEC standard curves as well as several vendor specific curves. Engineer can define which curves the solver is allowed to consider (to match utility's standard).
- Runtime costs – increased problem space. In theory could cause slowdowns. In practice, effect is minimal.

## Other Tunable Parameter Improvement

- Selective CT Ratio tuning (e.g., when changes already scheduled to be made).
- Alternative Pickup set strategies based on fault currents + multiplier.

# Overcurrent curve parameter selection

## Experimental Setup

- 9 bus / 18 relay system based on ASPEN OneLiner example.
- 1st experiment, U1-U5 curve, allowed line-end response: [0.5..1.25]sec. *(composable)*
- U4 chosen for all but one relay (R1, U3).
- Aggregate close-end EO response 2.99sec (27% reduction)

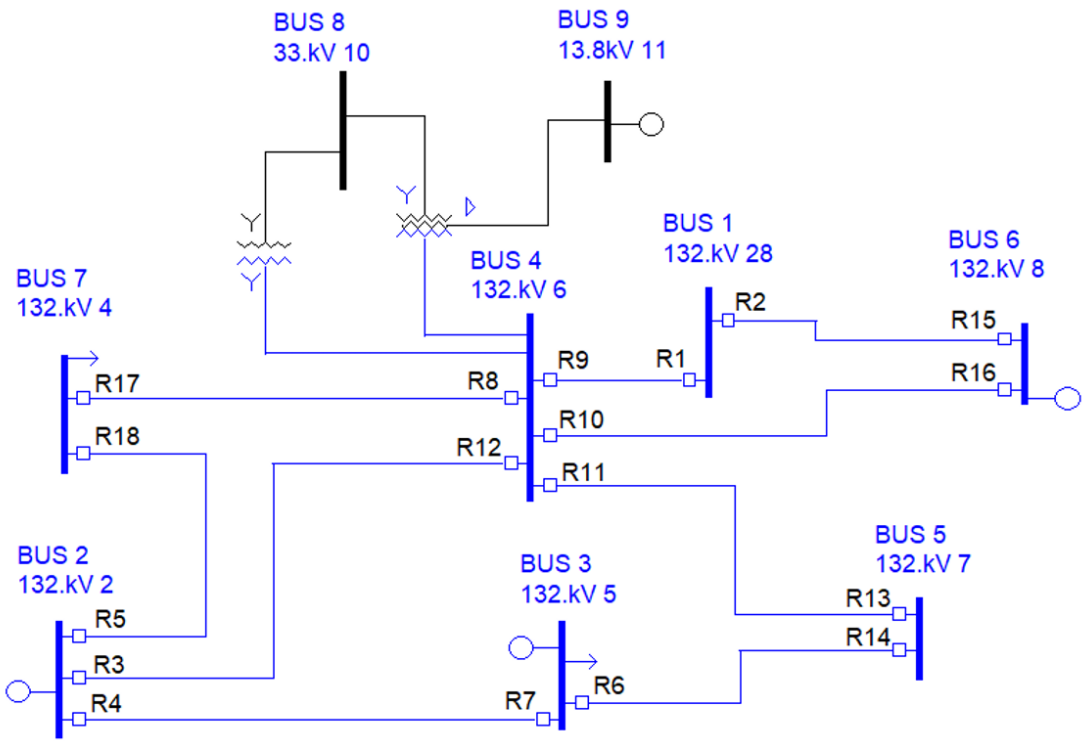



TABLE III
CALCULATED SETTINGS AND RESPONSE FOR 9-BUS SYSTEM (LINE-END CONSTRAINED) WITH CURVE SELECTION (U1-U5)

| Relay | $Curve$ | $I_{set}$ | $TMS$ | $T_{LE}$ (s) |
|---|---|---|---|---|
| R1 | U3 | 7.8169 | 3.7752 | 0.5000 |
| R2 | U4 | 6.7823 | 2.6467 | 0.5000 |
| R3 | U4 | 15.0000 | 0.7516 | 0.5000 |
| R4 | U4 | 15.0000 | 0.7983 | 0.5000 |
| R5 | U4 | 15.0000 | 2.0698 | 0.5000 |
| R6 | U4 | 15.0000 | 1.3798 | 0.5422 |
| R7 | U4 | 15.0000 | 0.5386 | 0.5000 |
| R8 | U4 | 9.2938 | 0.5073 | 0.7880 |
| R9 | U4 | 15.0000 | 3.8846 | 0.5000 |
| R10 | U4 | 15.0000 | 3.1128 | 0.5000 |
| R11 | U4 | 15.0000 | 0.9248 | 0.5000 |
| R12 | U4 | 15.0000 | 1.9444 | 0.5182 |
| R13 | U4 | 15.0000 | 1.7486 | 0.5000 |
| R14 | U4 | 15.0000 | 0.5000 | 0.5062 |
| R15 | U4 | 13.5263 | 0.5096 | 0.5000 |
| R16 | U4 | 11.3158 | 0.5260 | 0.6499 |
| R17 | U4 | 15.0000 | 0.7665 | 0.6136 |
| R18 | U4 | 12.0526 | 0.5000 | 0.5228 |

# Overcurrent curve parameter selection

## Experimental Setup

- 2st experiment, U1-U5 curve, allowed line-end response: [*0.7*..1.25]sec.
- If fixed U3 curve, no solution possible. Curve tuning critical.
- Aggregate close-end EO response remains *unchanged.*

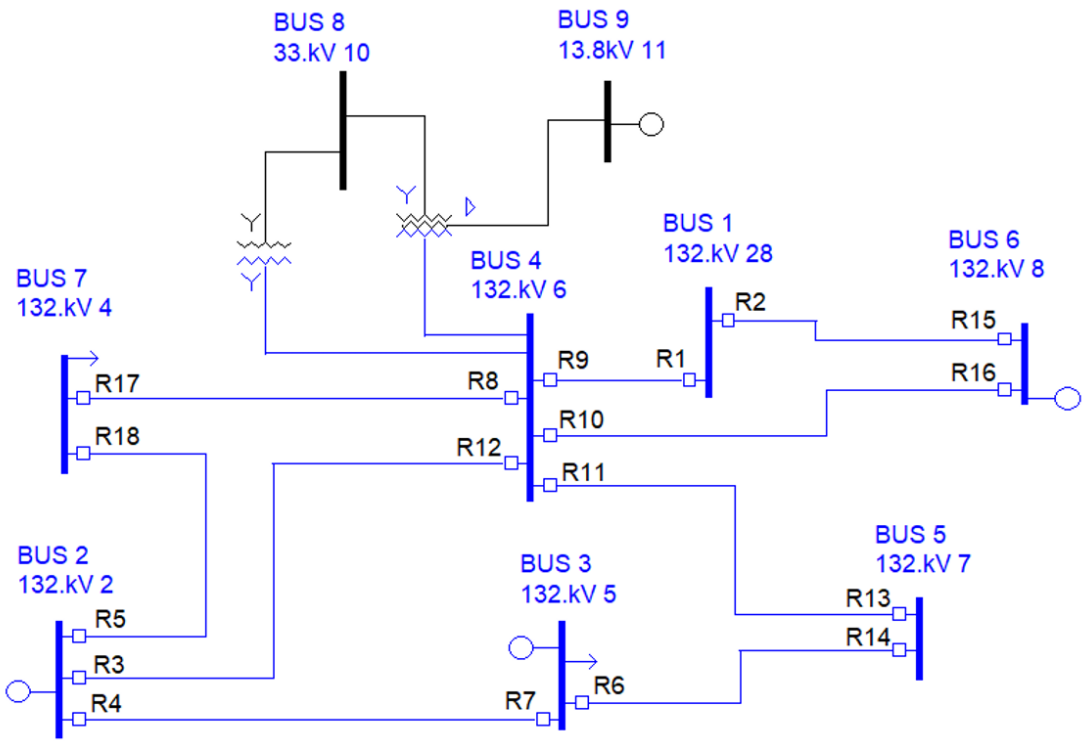



TABLE IV
CALCULATED SETTINGS AND RESPONSE FOR 9-BUS SYSTEM (LINE-END CONSTRAINED, $\geq 0.7$) WITH CURVE SELECTION (U1-U5)

| Relay | $Curve$ | $I_{set}$ | $TMS$ | $T_{LE}$ (s) |
|---|---|---|---|---|
| R1 | U2 | 7.8169 | 2.9733 | 0.7000 |
| R2 | U4 | 6.7823 | 3.7053 | 0.7000 |
| R3 | U4 | 15.0000 | 1.0522 | 0.7000 |
| R4 | U4 | 15.0000 | 1.1177 | 0.7000 |
| R5 | U4 | 15.0000 | 2.8977 | 0.7000 |
| R6 | U4 | 15.0000 | 1.7813 | 0.7000 |
| R7 | U4 | 15.0000 | 0.7541 | 0.7000 |
| R8 | U4 | 10.4786 | 0.5000 | 1.0581 |
| R9 | U3 | 15.0000 | 4.3670 | 0.7000 |
| R10 | U4 | 15.0000 | 4.3579 | 0.7000 |
| R11 | U4 | 15.0000 | 1.2948 | 0.7000 |
| R12 | U4 | 15.0000 | 2.6267 | 0.7000 |
| R13 | U4 | 15.0000 | 2.4481 | 0.7000 |
| R14 | U4 | 15.0000 | 0.6914 | 0.7000 |
| R15 | U4 | 15.0000 | 0.5623 | 0.7000 |
| R16 | U4 | 12.0526 | 0.5425 | 0.7796 |
| R17 | U4 | 15.0000 | 0.8905 | 0.7129 |
| R18 | U4 | 13.5263 | 0.5112 | 0.7000 |

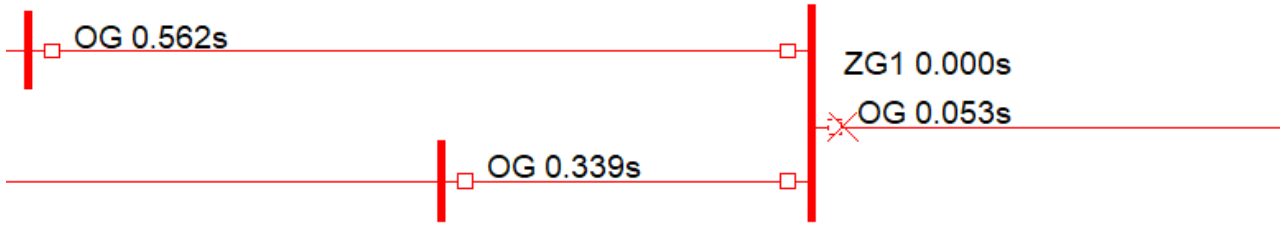
# Incorporation of Distance Elements

- Distance elements relatively easy to coordinate compared to OC elements. DS/DS can be done in similar but simplified manner as OC/OC.

- For paper, first address *heterogenous* element coordination to find undiscovered challenges.

- Investigated OC/DS coordination, namely incorporated local Zone 1 response selectively when candidate OC settings at primary and backup will fail to coordinate (i.e., short source lines). *Often used as a compromise by protection engineers.*

- Implementation – extract distance relay element response times from short circuit model and incorporate into problem formulation.



# Incorporation of Distance Elements

## Experimental Setup

- Real world, interconnected power system in Texas.
- U1 to U5 curves considered, allowed line-end response: [0.25..1.25] sec.
- Without Zone 1 response incorporated, system does not coordinate.
- With Zone 1 responses, optimal solution found in 3.45 seconds.

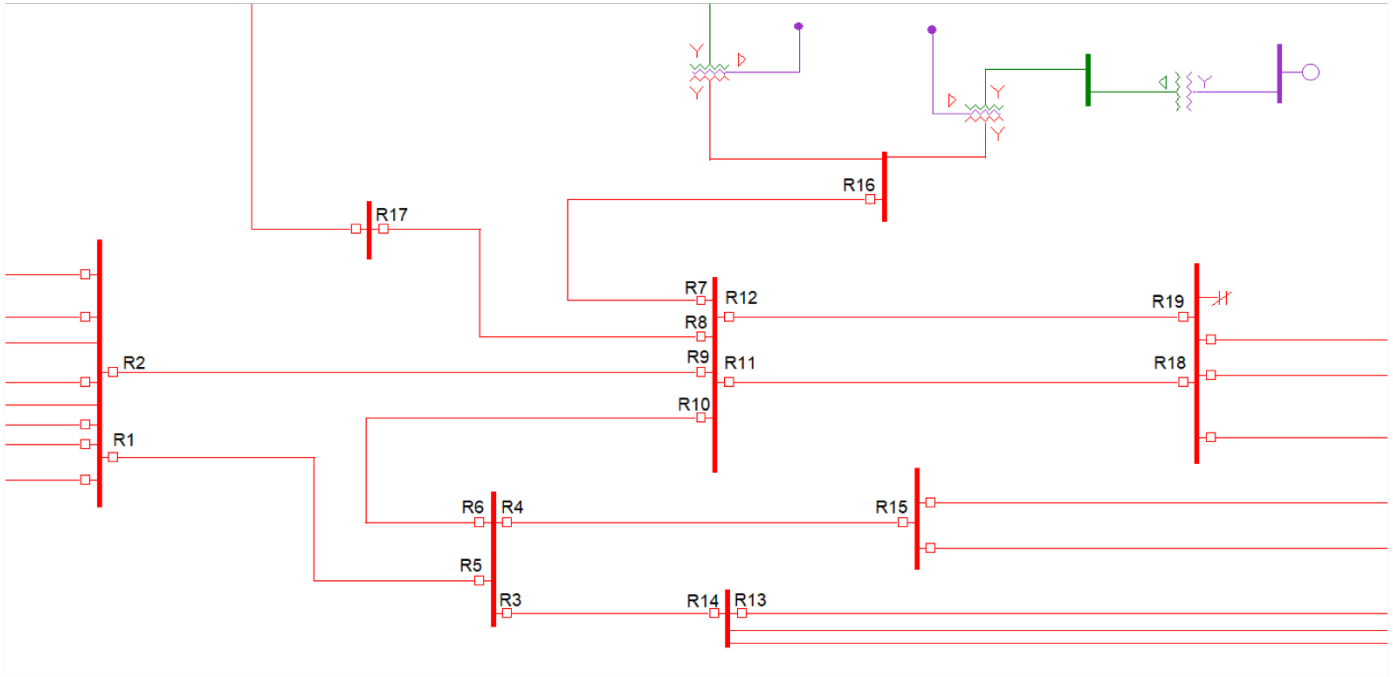
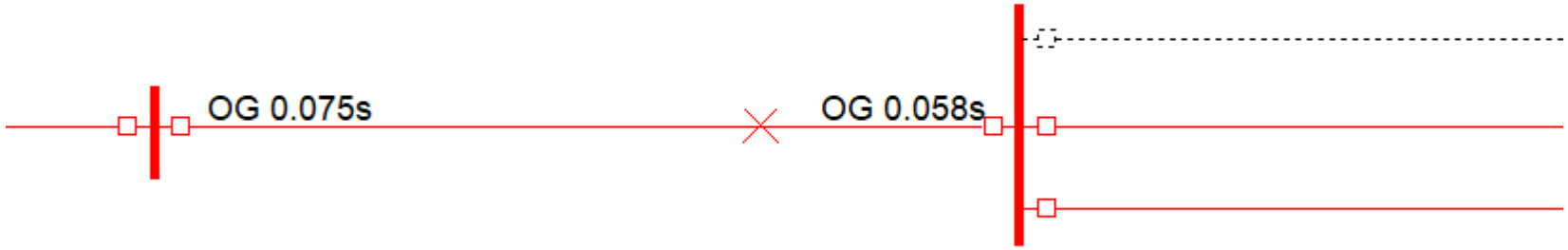



TABLE V
CALCULATED SETTINGS AND RESPONSE FOR REAL WORLD SYSTEM: LOCAL ZONE 1 RESPONSE INCORPORATED

| Relay | $Curve$ | $I_{set}$ | $TMS$ | $T_{LE}$ (s) |
|---|---|---|---|---|
| R1 | U4 | 10.4737 | 0.5331 | 0.2500 |
| R2 | U4 | 8.3252 | 0.5128 | 0.3533 |
| R3 | U2 | 16.0000 | 0.6185 | 0.4172 |
| R4 | U4 | 14.4211 | 1.0816 | 0.2500 |
| R5 | U4 | 16.0000 | 0.5000 | 0.2505 |
| R6 | U4 | 15.2105 | 0.5390 | 0.3280 |
| R7 | U1 | 1.0000 | 2.5028 | 0.3743 |
| R8 | U2 | 16.0000 | 0.5574 | 0.4558 |
| R9 | U4 | 14.0110 | 0.5000 | 0.2832 |
| R10 | U4 | 16.0000 | 0.5915 | 0.2500 |
| R11 | U1 | 1.0000 | 3.3123 | 0.6401 |
| R12 | U1 | 1.0000 | 3.3205 | 0.6476 |
| R13 | U1 | 1.0000 | 1.7103 | 0.2500 |
| R14 | U1 | 15.2105 | 1.6030 | 0.9416 |
| R15 | U1 | 1.0000 | 6.3453 | 1.1163 |
| R16 | U1 | 1.0000 | 7.8298 | 1.2102 |
| R17 | U1 | 1.0000 | 6.4754 | 1.1701 |
| R18 | U2 | 4.6165 | 0.5000 | 0.7269 |
| R19 | U2 | 4.0140 | 0.5000 | 0.5852 |

# Contingency Support

- Previous approach only considered normal conditions, not feasible for real-world use.
- Now implemented, expressed as additional constraints for each fault considered and each additional grid configuration.
- Uses topology analysis and fault simulation capabilities already developed [Corpuz 2017].



# Contingency Support

## Experimental Setup

- U1 to U5 curves considered
- allowed line-end response: [0.25..1.25] sec
- Zone 1 response considered.
- N-1 remote line contingencies.
- Exec time is 6 minutes (see paper for optimization strategies).

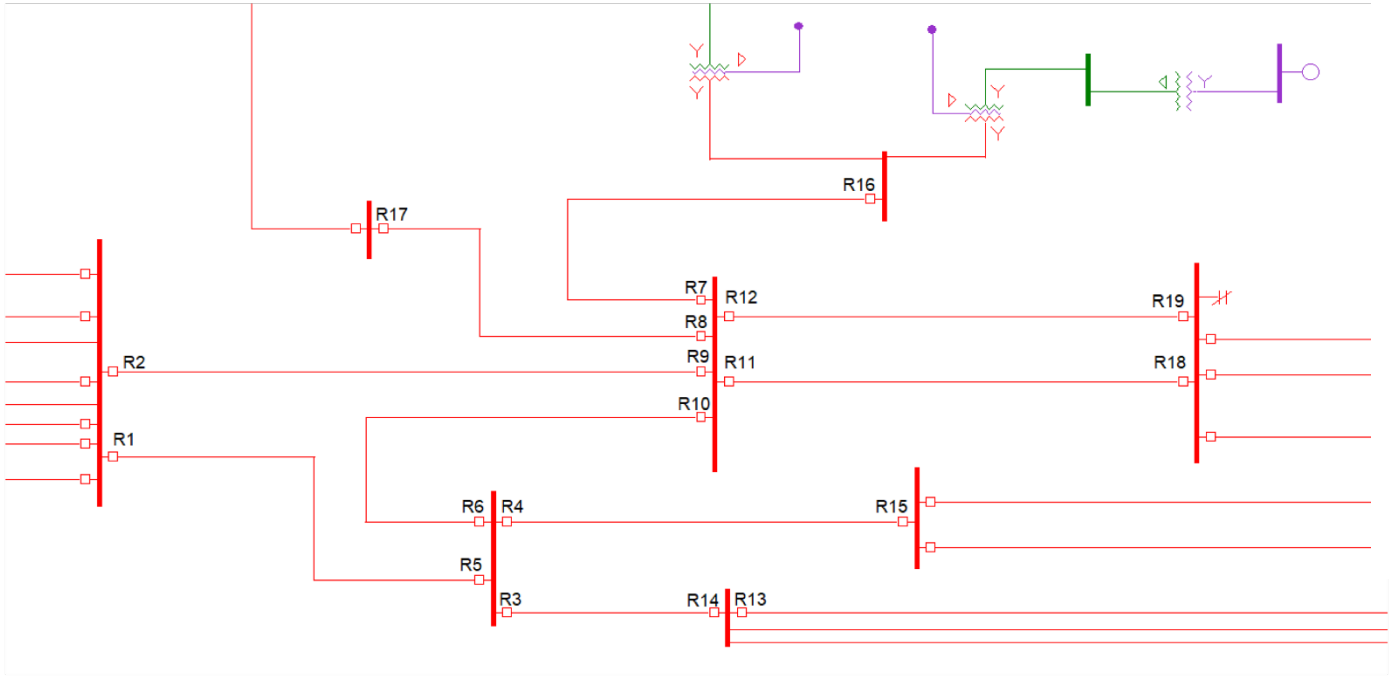



TABLE VI
CALCULATED SETTINGS AND RESPONSE FOR REAL WORLD SYSTEM: REMOTE LINE CONTINGENCIES AND LOCAL ZONE 1 RESPONSE INCORPORATED

| Relay | $Curve$ | $I_{set}$ | $TMS$ | $T_{LE}$ (s) |
|---|---|---|---|---|
| R1 | U4 | 12.8421 | 0.5000 | 0.3567 |
| R2 | U4 | 10.7253 | 0.5000 | 0.6048 |
| R3 | U4 | 16.0000 | 0.6116 | 0.3097 |
| R4 | U4 | 16.0000 | 0.8982 | 0.2500 |
| R5 | U4 | 7.3158 | 1.9739 | 0.2500 |
| R6 | U4 | 7.3158 | 2.1365 | 0.3381 |
| R7 | U4 | 13.6316 | 0.5352 | 0.2500 |
| R8 | U4 | 16.0000 | 0.7092 | 0.4559 |
| R9 | U4 | 12.8322 | 0.5273 | 0.2500 |
| R10 | U4 | 14.4211 | 0.7226 | 0.2500 |
| R11 | U4 | 5.3412 | 0.5000 | 0.2538 |
| R12 | U4 | 5.0941 | 0.5104 | 0.2500 |
| R13 | U4 | 15.2105 | 0.5290 | 0.2500 |
| R14 | U3 | 6.5263 | 3.7403 | 0.8043 |
| R15 | U2 | 1.0000 | 4.9669 | 0.9358 |
| R16 | U2 | 8.8947 | 1.7022 | 0.7302 |
| R17 | U2 | 4.9474 | 1.7023 | 0.7491 |
| R18 | U4 | 3.0054 | 0.5069 | 0.2500 |
| R19 | U4 | 2.9380 | 0.5000 | 0.2523 |

# When Full Coordination is not Possible.

## Allowing minimal violations when coordination not possible.

- Previous approach detected when full coordination was not possible but returned no approximate / proposed solution.
- Now, if no feasible solution found, solver attempts to find a solution with number and size (almost valid CTIs) that yields tuned settings.
- Reports the settings as well as the violations that were allowed to create them.

# Contingency Support

## Experimental Setup

- U1 to U5 curves considered
- line-end response: [0.5..1.25] sec
- Zone 1 response *not* considered.
- 2 line end time violations (worst off by 0.1241 seconds.
- 5 CTI violations (worst off by 0.0663 seconds / 4 cycles).

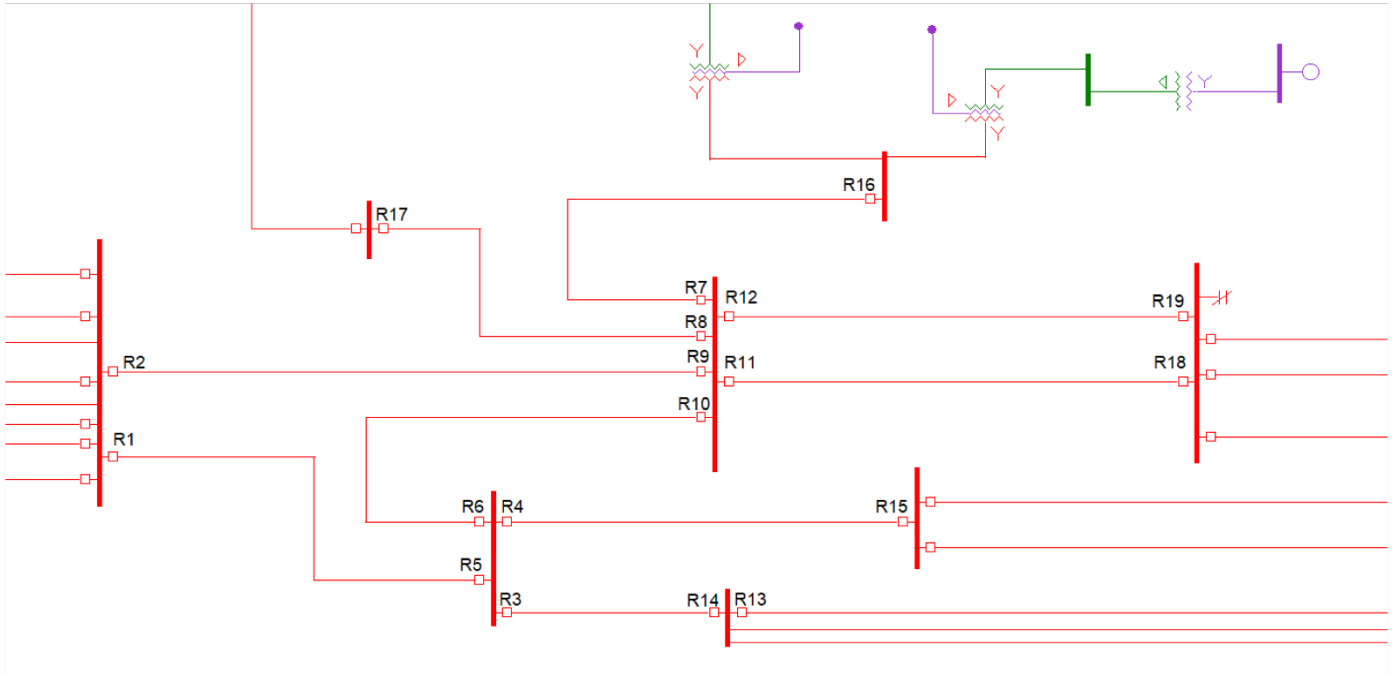



TABLE VII
CALCULATED SETTINGS AND RESPONSE FOR REAL WORLD SYSTEM: COORDINATED WITH MINOR VIOLATIONS

| Relay | $Curve$ | $I_{set}$ | $TMS$ | $T_{LE}$ (s) |
|---|---|---|---|---|
| R1 | U4 | 12.8421 | 0.5269 | **0.3759** |
| R2 | U4 | 10.1564 | 0.5000 | 0.5332 |
| R3 | U4 | 16.0000 | 0.9874 | 0.5000 |
| R4 | U4 | 16.0000 | 1.5628 | **0.4350** |
| R5 | U2 | 13.6316 | 0.9487 | 0.5000 |
| R6 | U4 | 16.0000 | 0.8162 | 0.5521 |
| R7 | U1 | 1.0000 | 3.3430 | 0.5000 |
| R8 | U3 | 15.2105 | 1.0677 | 0.5000 |
| R9 | U4 | 14.0110 | 0.8828 | 0.5000 |
| R10 | U2 | 16.0000 | 0.8524 | 0.5000 |
| R11 | U1 | 1.0000 | 4.4004 | 0.8503 |
| R12 | U1 | 1.0000 | 4.4113 | 0.8603 |
| R13 | U4 | 16.0000 | 0.9558 | 0.5000 |
| R14 | U3 | 8.8947 | 3.6903 | 1.1911 |
| R15 | U2 | 1.0000 | 7.9612 | 1.5000 |
| R16 | U2 | 10.4737 | 2.8255 | 1.5000 |
| R17 | U1 | 1.0000 | 8.3010 | 1.5000 |
| R18 | U2 | 3.6648 | 0.5412 | 0.5000 |
| R19 | U1 | 3.6967 | 1.1860 | 0.6048 |

TABLE VIII
CTI VIOLATIONS FOR REAL WORLD MODEL WITH LINE-END FAULT RESPONSE TIME $\geq$ 0.5s

| Primary Relay | Fault Type | CTI |
|---|---|---|
| R1 | Close In EO | 0.2767 |
| R1 | Close In | 0.2637 |
| R1 | Interm. 10% | 0.2988 |
| R2 | Close In EO | 0.3000 |
| R2 | Close In | 0.2933 |
| R4 | Close In | 0.3247 |

# Minimizing Violations

- The ability to identify and minimize violations to find a near full coordination is an important breakthrough.

- The autotuner can now be incredibly useful in the cases that matter most (hardest).

- Future work – all violations are not equal.
  Allow prioritizing of some contingencies over others (N-0 -> N-1 -> N-2).

# Future Work

- Still in prototype form.
  - Test and adapt to larger grids.
  - Expand support for distance elements
  - More critically evaluate against historical studies and refine implementation.

- Develop graphical user interface
  - Ease specification of coordination specification.
  - Visualization of candidate solutions.

- Model the execution time of the software and optimize.
  - Employ larger scale parallelism (currently core multi-threading) to reduce compute + analyze loop.

# Questions?

NATHAN THOMAS, Ph.D.
Software & Automation

LUKE HANKINS, P.E.
Protection Engineer

JOE PEREZ, P.E.
Founder – Protection Engineer

**Presented at the 74th Annual Texas A&M Relay Conference**