Pacific Northwest
NATIONAL LABORATORY

PNNL-34604-1

Distributed Rules-Based Deconfliction of ADMS Applications

Part 1: Requirements & Decomposition

March 2023

Alexander Anderson
Todd Wall
Subramanian (Mani) Vadari
Andrew Reiman

U.S. DEPARTMENT OF ENERGY

Prepared for the U.S. Department of Energy
under Contract DE-AC05-76RL01830

DISCLAIMER

PACIFIC NORTHWEST NATIONAL LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC05-76RL01830

Printed in the United States of America

**Available to DOE and DOE contractors from
the Office of Scientific and Technical
Information,
P.O. Box 62, Oak Ridge, TN 37831-0062**
www.osti.gov
**ph: (865) 576-8401
fox: (865) 576-5728
email: reports@osti.gov**

**Available to the public from the National Technical Information Service
5301 Shawnee Rd., Alexandria, VA 22312
ph: (800) 553-NTIS (6847)
or (703) 605-6000
email: info@ntis.gov
Online ordering: http://www.ntis.gov**

PNNL-34604-1

Distributed Rules-Based Deconfliction of ADMS Applications

Part 1: Requirements & Decomposition

March 2023

Alexander Anderson
Todd Wall
Subramanian (Mani) Vadari
Andrew Reiman

Prepared for
the U.S. Department of Energy
under Contract DE-AC05-76RL01830

Pacific Northwest National Laboratory
Richland, Washington 99354

Summary

This work introduces a conceptual framework and preliminary methodology for resolving conflicting device control commands issued by advanced power applications. The methodology is designed to serve as one of multiple possible alternative implementations for the numerical component of the Deconfliction Pipeline introduced in [1]. The Deconfliction Pipeline is a framework for deconflicting multiple applications that intercepts control commands issued by advanced applications to (sometimes the same) field equipment, identifies possible conflicts, and generates new controls and/or setpoints. It is anticipated that future operational implementations of the Deconfliction Pipeline will use a combination of the rules-based methodology (introduced in this work), application cooperation, and global optimization.

The rules-based deconfliction methodology focuses on using a combination of technical, economic, environmental, and social rules to guide selection of a suitable set of equipment controls and/or setpoints that are acceptable to the distribution system operator and do not violate real-time operational constraints. The methodology can be implemented as constraints on a global optimization problem or as a decision criteria as part of a multi-criteria decision-making framework. The second approach is well-suited for use with distributed applications and distributed computing, and thus, will be the focus of two complementary reports, *Distributed Rules Based Deconfliction, Part 1 and Part 2*.

Rules Based Deconfliction: Part 1 applies a Grid Architecture approach to characterize the domain problem and introduces the context, requirements, and methods for handling the deconfliction problem. Key concepts of the Laminar Coordination Framework and Variable Grid Structures are applied to define a conceptual framework for formulating the deconfliction problem. The deconfliction problem is then decomposed into a distributed optimization problem based on the concept of quasi-static grid segments, which form independent distributed areas for control and coordination.

Distributed Rules Based Deconfliction, Part 2 details the various criteria, rules, and processes by which setpoints are analyzed, ranked, and ultimately selected. It introduces an initial set of technical, economic, and environmental criteria for generating a deconfliction solution that meets the requirements of the distribution system operator. It also introduces specific qualitative rules that are used as part of the deconfliction methodology to eliminate non-viable setpoint alternatives. This work also details how the deconfliction optimization problem is converted into a ranking of individual discrete setpoints, scored by specific decision criteria. Several multi-criteria decision-making frameworks are examined with the simple multi-attribute rating technique exploiting ranks (SMARTER) recommended as a simple implementation alternative that aligns with the steps of the rules-based deconfliction methodology.

Acknowledgments

The authors wish to acknowledge the contributions, input, and invaluable discussion provided by the other members of the application deconfliction effort, including Shiva Poudel, Monish Mukherjee, Orestis Vasios, Tyler Slay, and Gary Black. The authors also wish to thank Bora Akyol for his input on specification of control priorities and the core members of the GridAPPS-D leadership and platform teams, especially Jim Ogle, Poorva Sharma, Craig Allwardt, and Jonathan Barr for their input and concepts used for definition of the distributed rules-based deconfliction architecture.

Acronyms and Abbreviations

ADMS	Advanced Distribution Management System
AGC	Automatic Generation Control
AOR	Area of Responsibility
AMI	Advanced Metering Infrastructure
AVR	Automatic Voltage Regulation
CIM	Common Information Model
CVR	Conservation Voltage Reduction
DER	Distributed Energy Resource
DMS	Distribution Management System
DR	Demand Response
ESB	Enterprise Service Bus
FLISR	Fault Location Isolation and Restoration
HV	High Voltage
IEC	International Electrotechnical Commission
IT	Information Technology
LTC	on-Load Tap Changer
MADM	Multi-Attribute Decision Making
MAUT	Multi-Attribute Utility Theory
MCDM	Multi-Criteria Decision Making
MODM	Multi-Objective Decision Making
OMS	Outage Management System
OT	Operations Technology
PCC	Point of Common Coupling
SCADA	Supervisory Control and Data Acquisition
SMART	Simple Multi-Attribute Rating Technique
SMARTER	Simple Multi-Attribute Rating Technique Exploiting Ranks
VVO	Volt-Var Optimization

Contents

1.0 Introduction

Decarbonization, electrification, and grid modernization efforts are rapidly transforming electric distribution networks, with significant increases in penetration levels of distributed energy resources (DERs), including distributed generation, battery storage, electric vehicles, and controllable customer-owned resources and supported by increased electric load (consumption). Additionally, numerous new data streams are becoming available, including advanced metering infrastructure (AMI) systems, grid-edge devices, internet of things (IoT) devices, smart inverter controllers, intelligent reclosers, and micro-phasor measurement units. To leverage these new devices and data streams, distribution utilities are investigating the need to invest in new advanced power applications as part of advanced distribution management systems (ADMS) and distributed energy resource management systems (DERMS) implementations.

Within this new paradigm, the historically challenging data integration problem is complicated further by the emergence of application conflict. Application conflict arises as each new advanced power application pursues its own set of objectives and may send control commands to change the state of these devices. Unless these applications have been carefully integrated and tested as part of a single vendor suite, it is likely that different applications may issue competing setpoints to the same device. Without a robust deconfliction process, multiple mis-operation scenarios are possible, including 1) devices responding to setpoints arbitrarily, 2) oscillating system behavior as conflicting setpoints arrive asynchronously, and 3) applications completely failing to achieve their individual objectives.

This work introduces a conceptual structure for a rules-based deconfliction methodology as one of three possible numerical methods that can be implemented in the Deconfliction Pipeline proposed by [1]. The concepts and tools of Grid Architecture are leveraged extensively, along with a set of design principles that recognize that power systems are not simply an electric circuit. Instead, the electric grid comprises a network of structures involving control, coordination, communications, sensing, and data management. This network of structures is used to define substructure specific to the application deconfliction problem and sequential steps used to derive a locally optimal set of deconflicted device setpoints.

Development of the rules-based deconfliction methodology is divided into two parts. This first document introduces the requirements, context, and methods for decomposing the deconfliction problem, with a detailed list of operational and functional requirements that must be satisfied by conceptual implementation and thirty specific qualitative rules that are used as part of the deconfliction methodology are outlined in the second document.

The remainder of this document is organized as follows: Section 2 introduces the guiding principles for the rules-based deconfliction methodology, which are grounded in control room operations procedures and the design principles of Grid Architecture. Section 3 applies a set of Systems Engineering methods to define the key operational and functional requirements of the deconfliction methodology. Section 4 introduces a distributed deconfliction architecture based on separation of the information technology (IT) data bus from the operations technology (OT) control bus for use with both centralized and distributed applications.

2.0 Application Deconfliction Context & Principles

2.1 Current Distribution System Operations

Within most distribution utility control rooms, operational tasks are generally divided between the distribution system operator, system dispatcher, and field crews [2]. The system operator is responsible for operating remote-controllable equipment through a supervisory control and data acquisition system (SCADA), including substation breakers, on-load tap changer (LTC) transformers, and intelligent reclosers. The system operator is also responsible for monitoring system parameters, such as the temperature and loading of high-voltage (HV) substation transformers, and coordinating operational procedures, such as load rollover.

The system dispatcher is responsible for coordinating with the system operator and field crews using a wallboard map of the utility service territory, as shown in Figure 1. The system dispatcher requests changes to equipment settings and creates switching orders in response to unplanned outages, planned maintenance tasks, and energization of new customers. The system dispatcher maintains three-way verbal communication with field crews via telephone and/or radio.

Figure 1: Operations wallboard of a municipal distribution utility with equipment positions and outages marked by colored flags on a geospatial map (courtesy Alka Singh & Rhonda Schennum)

The field crews are responsible for performing truck rolls to identify downed conductors, blown fuses, and failed equipment. They are also responsible for executing switching orders and performing a variety of maintenance tasks, such as changing settings of manually controlled equipment, seasonal re-phasing of customers, and hot-line work (cuts and jumpers) on energized conductors.

2.2 Emerging Data-Rich Controls and Operations

Grid modernization and distribution automation efforts are driving a significant increase in the observability and controllability of distribution networks. Installation of SCADA-enabled reclosers, voltage regulators, and dispatchable DERs are motivating electric utilities to invest in new software systems, including ADMS[1] and DERMS. These new software tools introduce a range of new functionalities and advanced power applications designed to increase the reliability, resiliency, and efficiency of the distribution network [3]. Common applications found in most vendor suites include [4]

- Volt-Var Optimization (VVO)
- Fault Location Isolation and Restoration (FLISR)
- Conservation Voltage Reduction (CVR)
- Demand Response (DR)
- DER control and coordination (optimization and dispatch)
- Field crew location & optimal routing.

Introduction of these tools into distribution utility control rooms is transforming the role and workflows of system operators and system dispatchers, with increasing automation of common tasks, such as creation and execution of switching orders. In modernized control rooms (Figure 2),

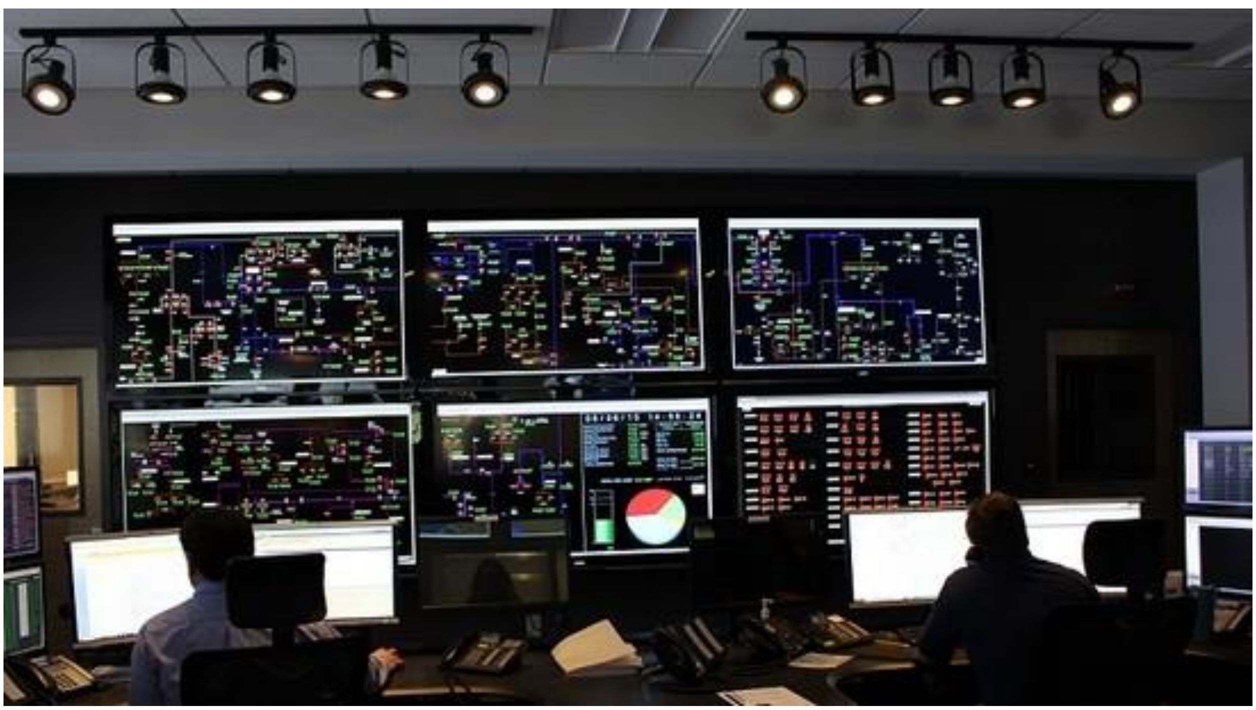

Figure 2: Modernized distribution system control room with SCADA and OMS, taken from [19].

[1] An ADMS is typically defined as the combination of SCADA, an outage management system (OMS), and any additional distribution management system (DMS) applications.

operators now interact largely through computer-based interfaces with ADMS and DERMS applications. Early distribution automation tools (such as FLISR) required manual approval of all equipment control actions by the operator, but these tools are now being granted increasingly more direct control of field devices [2].

With the introduction of more advanced applications with direct control of field devices, often from different vendor suites, an emerging problem is that of conflicting control commands issued by separate applications. This is a new problem that has not been seen previously in either transmission or distribution operations due to traditional reliance on manual operation of devices and direct feedback loop controls, such as automatic generation control (AGC) and automatic voltage regulation (AVR). An initial characterization of the deconfliction problem is introduced by [1].

2.3 Conflicts between Power Applications

Conflicts between applications can arise for a variety of reasons, including usage of incompatible control strategies, conflicting numerical objective functions, limited number of control options, and differing system-level mandates [1]. This can result in two possible manifestations of application conflict in the real-time operations timeframe. The first is direct control command collisions, where two or more applications issue different setpoints for the same controllable device, either simultaneously or within a subsequent application iteration. An example of a direct command collision is a VVO application sending a control message to raise the tap of a poletop voltage regulator and a CVR application sending a message to lower the tap of the same regulator. The second manifestation of application conflict is physics-domain collisions of control commands that counteract each other in terms of net result on power flow or other physical parameters. An example of such a collision involves an optimal DER dispatch application that sends commands to dispatch reactive power of batteries (to increase voltage) and a VVO application that subsequently opens capacitor banks (to decrease voltage). These applications do not conflict directly from a device setpoint perspective but are issuing commands that counteract each other and need to be deconflicted. Either of these command collisions result in a setpoint mismatch for the controllable devices within the distribution network, resulting in multiple mis-operation scenarios, including devices responding to commands arbitrarily, oscillating system behavior as conflicting setpoints arrive asynchronously, and applications completely failing to achieve their objective.

Traditional approaches to data integration and application integration focus on data exchange and ensuring that applications are able to communicate effectively with the SCADA system, legacy application suites, field devices, and other software components (such as network model databases) [5]. However, data integration and information exchange between different suites of applications does not necessarily guarantee there will be no conflict between individual applications or device controllers. Likewise, numerous numerical methods exist for resolving conflicts between individual objective functions, but direct implementations present several challenges. The first is the requirement that all applications share their objective function formulation, which may be unacceptable to some ADMS vendors and developers of "black-box" software using novel control formulations that represent their commercial competitive advantage. Without full knowledge of all objectives, constraints, and solution variables used by all applications on the same distribution network, it is not possible to formulate a numerical problem to find globally optimal device setpoints. The second challenge is that even if all application parameters are known, direct implementation of any of the methods listed above requires development of a set of custom adapters for each application. Introduction of a new application after completion of the previous application integration effort would require development of new custom formulation and set of software adapters.

2.4 Grid Architecture Approach

The application deconfliction methodology introduced in this report will heavily leverage multiple concepts from the domain of Grid Architecture, which synthesizes numerous concepts from system architecture, theory of networks, and controls engineering [6]. The focus of Grid Architecture is to specify the components, structures, and externally-visible properties of the modernized electric power system.

Grid Architecture introduces several techniques for characterizing the technical domain and decomposing complex problems, including the entity-relations diagrams [7], variable structure grids [8], and the Laminar Coordination Framework [9]. These architectural components are combined with a set of best practices and design principles [6] [10], including:

- **Layered Decomposition**: Complex problems can be decomposed into smaller subproblems with solutions orchestrated through coordination nodes using a structure such as the Laminar Coordination Framework.

- **Avoiding Tier Bypassing**: Data flows, controls signals, or coordination/dispatch paths should not bypass any tier of the power system structural hierarchy (e.g., a transmission operator should not be able to dispatch a microgrid resource while bypassing the distribution system operator and microgrid controller).

- **Avoiding Hidden Coupling**: Careful analysis of control and coordination structures is performed to avoid two or more control systems with partial views and possibly unrelated individual objectives. These conflicts may even come from different organizational entities with applications sending conflicting control signals to a device.

- **Avoiding Latency Cascading**: Serial data flows between different systems and organizations are assessed, managed, and minimized as they can quickly introduce excessive latencies, negatively impacting relevance and validity of real-time data.

These principles will be used to guide the design and formulation of the deconfliction methodology presented in this paper.

2.5 Separation of IT Data Bus and OT Control Bus

The rules-based deconfliction methodology is based on a novel architecture introduced by [11], which separates the information technology (IT) data bus from the operational technology (OT) control bus. Within this structure, it is possible to integrate centralized, distributed, and decentralized controllers while separating operational and non-operational data streams, as illustrated in Figure 3. In an operational context, the IT Data Bus and OT Control Bus should be

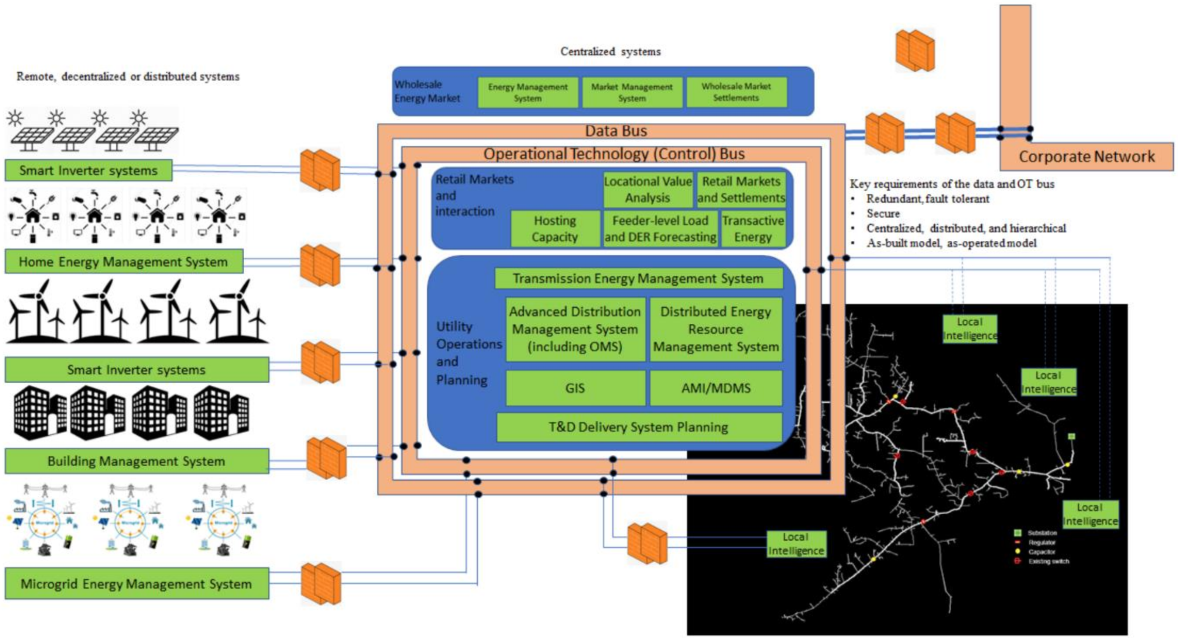

Figure 3: Conceptual separation of data flows onto an IT Data Bus and OT Control Bus, taken from [11]

highly secure, redundant, and fault-tolerant, with numerous mechanisms in place to ensure data privacy and security.

The IT Data Bus is largely similar to the Enterprise Service Bus defined in IEC standard 61968-1 [12] for information exchange between network operations, asset management, customer management, network model management, operations planning, and market settlement. The IT Data Bus manages slow-changing and static data related to operational and non-operational business processes. Examples of data flows though the IT Data Bus include delivering power system model information (such as equipment configurations, impedances, and topology) to users, applications, and shared services, as shown in Figure 4. Data exchange through the IT Data Bus also includes handling of non-operational data (such as ownership information, DER characteristics, geospatial locations, and controllability of equipment). The IT Data Bus is also responsible for hosting the data manager(s) needed to keep track of changes to the power system model with logging of change source and time stamp [11].

Meanwhile, the concept of an OT Control Bus is relatively new and is focused on providing data integration of field telemetry and control coordination in real-time or near-real-time operations [11]. The OT Control Bus is responsible for handling of equipment control commands, aggregation, and correlation of incoming data (from both SCADA and grid-edge devices), and handling of critical messages (such as alarms). This separation provides several advantages from an operational perspective:

- Separate the two networks in terms of securing just the mission critical information passing instead of everything. Reduces payload of data being transferred as well as the number of non-secure systems to connect with.
- Allows the OT bus to be able to transfer data from one location to another at high speeds.

- Allows independent distributed points of intelligence to talk with each other instead of all communication going to a central location and back.

From a deconfliction perspective, the biggest benefit is that with all controls flowing through one (dual-redundant) OT bus, logic can be embedded into the control bus to reduce opportunities for deconfliction – instead of in the algorithms.

2.6 Deconfliction Pipeline

In response to the growing need for a standardized, replicable approach to deconfliction of advanced applications, the concept of the Deconfliction Pipeline was introduced in [1]. The Deconfliction Pipeline conceptually provides a bridge between setpoints issued by applications hosted on the IT data bus and physical devices connected to the OT control bus through a communication network, as shown in Figure 4.

The Deconfliction Pipeline is responsible for generating new setpoints for field devices (the "deconfliction solution") by intercepting command issued by "black-box" proprietary applications, "naïve" power applications (which repeatedly issue a conflicting setpoint request), and "intelligent" power applications (which share deconfliction-related parameters in a cooperative manner). The intercepted setpoints are fed into the Setpoint Processor, which combines the individual setpoints requests from each application and combines them into an array of setpoints. The feasibility of these setpoints (from the perspective of physical device limits and operational constraints) is checked by the Feasibility Maintainer. To help detect direct command collision or a physics-based collision (defined in Section 2.3), the Deconfliction Pipeline builds an array of conflicting setpoints known as the Conflict Matrix. The Conflict Matrix is formulated as an array of setpoints with the rows corresponding to the controllable attributes of field devices and columns corresponding the individual requests from each application. The process for formulating and maintaining the Conflict Matrix is described in detail by [1].

The Deconfliction Pipeline then passes the Conflict Matrix to the Deconflictor service, which is responsible for determining an acceptable set of non-conflicting setpoints informed by current operating conditions and the set of user-specified and application-specified priorities and preferences. The Conflict Identification process analyzes the Conflict Matrix to identify any conflicts between setpoints. Detection of a conflict triggers the Deconfliction Methodology to solve a numerical problem and determine an acceptable deconfliction solution. The numerical solution is arranged into a Resolution Vector of new setpoints. The scope of this work will be strictly limited to formulation of one possible numerical technique inside the Deconfliction Methodology block of Figure 4.

The Setpoint Validator contains a set of operational safety-checks to ensure that the resolved vector of new device setpoints does not violate any ongoing work orders or safety limitations not included in the deconfliction methodology. The rules used for setpoint validation may need to be adapted to the individual utility's operating procedures for coordination of common tasks, such as hot-line work, outage planning, and shift turnover. Finally, the Device Dispatcher is responsible for converting the validated setpoints into a set of dispatch commands that are sent to devices using the appropriate communications structure.

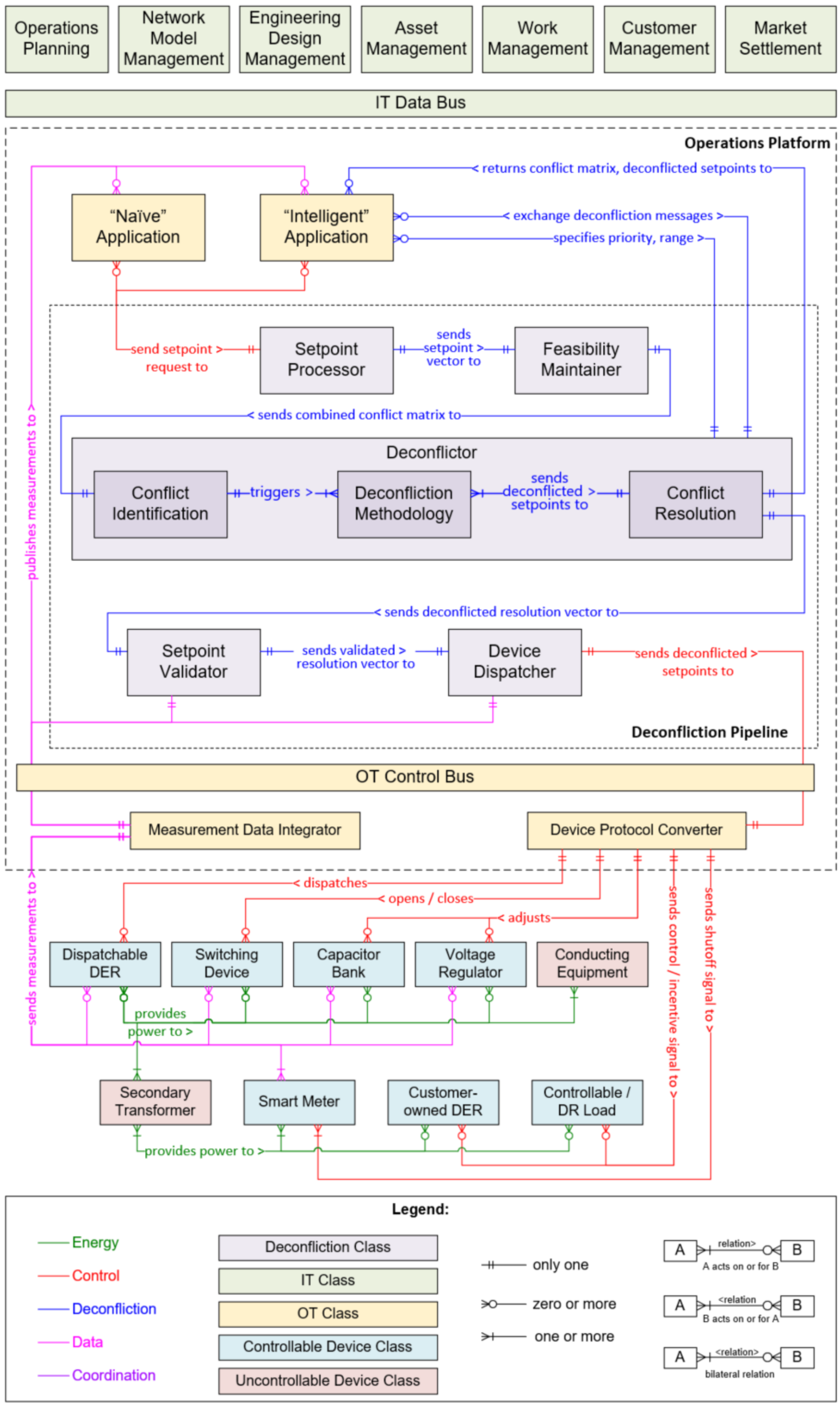

Figure 4: Deconfliction Pipeline for identifying and resolving conflicting setpoints issued by advanced power applications

2.7 Distributed Coordination and Decision-Making

A fully distributed decision-making approach is based on the structures of the Laminar Coordination Framework (LCF) [9], [13]. This framework reduces the complexity of the application deconfliction methodology via a formulation of a global optimization problem with coupled constraints. The LCF decomposes the coupled constraints into individual subproblems handled by layered coordination nodes, as shown in Figure 5. The coordination nodes are responsible for distributed processing of the local optimization subproblem and exchanging coordination signals across a message bus.

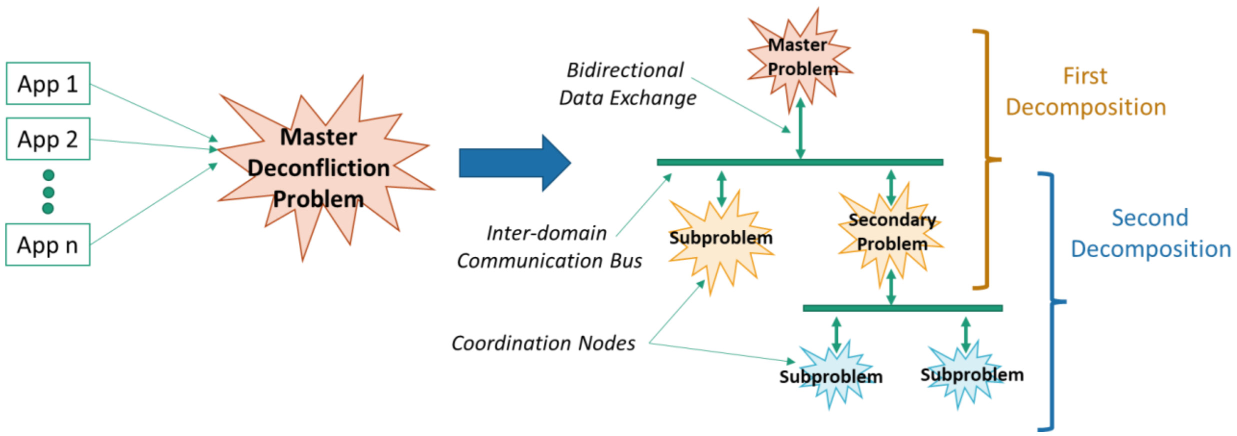

Figure 5: Decomposition of the application deconfliction problem using the Laminar Coordination Framework into subproblems handled by layered coordination nodes.

Notable characteristics and benefits [9] of problem decomposition using the Laminar Coordination Framework include

- **Extensibility**: The framework can be adapted to match existing grid structures
- **Scalability**: Coordination nodes do not need to be aggregated, avoiding communication scalability due to depth of the coordination chain
- **Boundary Deference**: concrete boundaries and interfaces are defined for compartmentalization of individual subproblems
- **Local Objective Support**: individual applications are allowed to seek selfish optimization objectives within each local subproblem
- **Local Constraint Support**: local constraints (such as thermal limits) are handled in a distributed fashion by the corresponding coordinator node without the need to exchange constraints across different layers

Although the original formulation of the Laminar Coordination Framework [9] is built from an optimization perspective, partial and/or missing information about application objectives, constraints, and requested setpoints hinders direct formulation of a distributed optimization problem. Instead, the deconfliction methodology must be able to infer the deconfliction context and requirements from real-time power system conditions and the preferences of the distribution system operator. The inference and decision-making processes are assigned to the appropriate coordination nodes for each distributed control area to provide scalability across the distribution network, which for a typical electric utility will contain hundreds of individual feeders with thousands of controllable devices.

3.0 Rules-Based Deconfliction Objectives and Context

The domain of Systems Engineering offers several tools and design lifecycles to ensure that a technical solution methodology is appropriate for the problem, satisfies key requirements, and can be integrated with existing solutions. One useful visual tool is the Triumvirate of Conceptual Design [14], shown in Figure 6, which illustrates the set of requirements and interrogatives that must be asked prior to commencing the engineering design process and are answered in the subsections below.

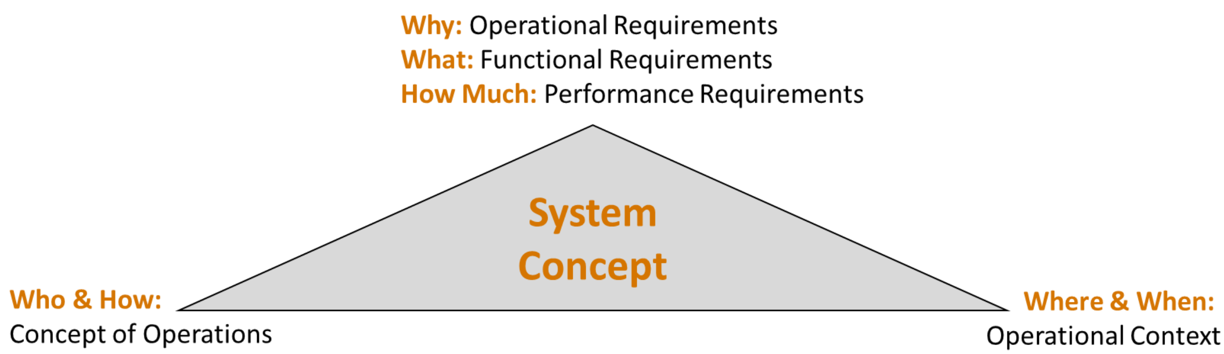

Figure 6: Triumvirate of Conceptual Design [14], which formulates seven interrogatives that form the basis of the requirements, operational context, and concept of operations.

3.1 Operational Requirements

Operational requirements refer to the mission and purpose of the deconfliction methodology (i.e. *why* it is needed). It has been previously identified that the generalized deconfliction methodology must

1) Be informed by the experiences and operational requirements of the utility industry
2) Ensure that system-level operational constraints are satisfied
3) Minimize restrictions placed on the design space of applications
4) Allow multiple applications to operate concurrently
5) Incentivize flexible and/or cooperative behavior by applications
6) Support centralized and distributed applications

3.2 Functional Requirements

Functional requirements describe *what* tasks need to be performed and *what* functionalities the methodology should provide. The rules-based deconfliction methodology should

1) Use standardized message structures, ontologies, and information models (such as CIM)

2) Understand preferences, priorities, and setpoints specified by applications
3) Allow assignment of exclusive control of individual devices to particular applications
4) Enable applications to specify a range of setpoints that are acceptable and should be used as bounds on the solution space
5) Recognize and enforce utility areas of responsibility (AOR) such that applications cannot control devices outside their service territory or AOR
6) Be aware of which devices can be controlled remotely by applications or manually by field crews
7) Intercept setpoint requests and control commands published by applications that may cause device mis-operation
8) Create new device setpoints that are acceptable to all stakeholders and operational entities
9) Create new device setpoints that consider the selfish optimization goals of individual applications to the greatest extent known
10) Include a failback mechanism to ensure safe operations during a range of failure scenarios, including loss of communications, server failures, and numerical solution divergence
11) Ensure that all deconflicted setpoints will not exceed system or device operating limits
12) Ensure that all deconflicted setpoints will not result in unsafe operation conditions
13) Ensure that all deconflicted setpoints will not result in damage or rapid degradation of physical utility-owned assets or customer equipment

3.3 Concept of Operations

A key prerequisite to developing an effective deconfliction methodology is understanding the interactions between stakeholders, applications, devices, and other actors involved in the application deconfliction process. The concept of operations (CONOPS) describes who is involved and how those entities interact in a structure manner. A Grid Architecture tool that can be used to capture these interactions is the entity-relationship diagram, which lists classes of entities and the set of unilateral and bilateral relationships between each entity.

The entity-structure diagrams for distribution systems presented by [6], [15] are extended down to the device level in Figure 6. Structures not directly related to the Deconfliction Pipeline (such as market agreements and regulatory entities) are omitted for visual clarity. A single class textbox can represent multiple instances of a device or application. The number of entities involved in a single relationship is indicated by the arrow type.

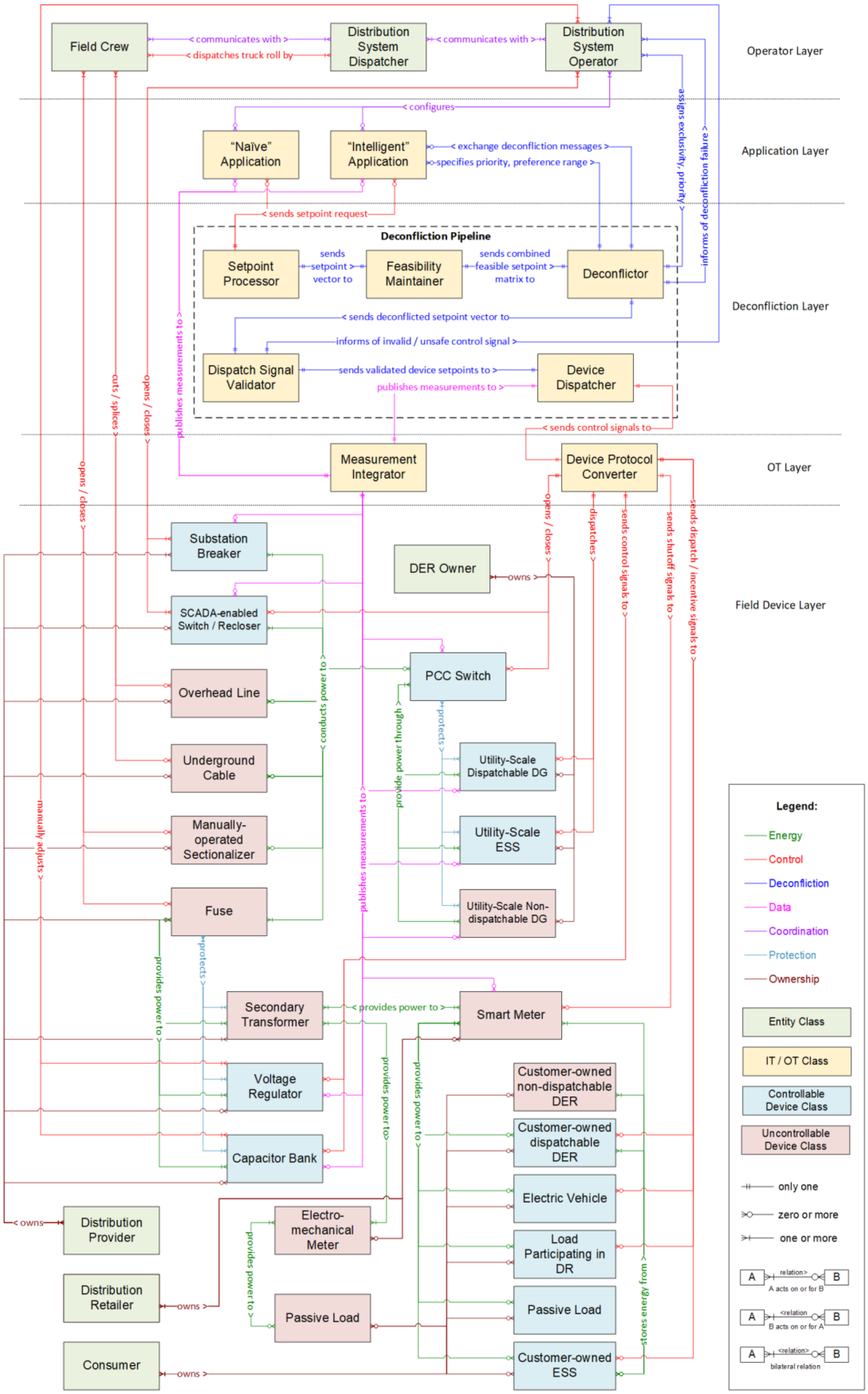

Figure 7: Entity structure diagram illustrating relationships between key stakeholders, IT / OT processes, and field devices

At the highest level of Figure 6 are the human stakeholders that directly interact with the field devices, applications, and deconfliction solution. They are able to take direct control of the devices through SCADA or manual control, as well as specify parameters to tune the deconfliction solution. At the next level are the advanced power applications and Deconfliction Pipeline introduced in [1]. The deconfliction pipeline is responsible for intercepting control commands, deconflicting them, and issuing new setpoints. The deconflicted setpoints are then converted to the correct communications protocol (at the OT layer) and broadcast to field devices. The structure diagram applies to centralized, distributed, and decentralized control architectures, with the same agents and interactions occurring inside each local control area. In distributed architectures, distributed Deconfliction Pipeline instances may coordinate and share decisions with each other using a hierarchical or peer-to-peer structure.

3.4 Operational Context

The final set of interrogatives are used to define the operational context, which explains *where* and *when* the system will be used. The rules-based deconfliction methodology described in this report lies inside the Deconflictor Block, which can be expanded in more detail, as shown in Figure 7. The three subprocesses focus on conflict identification from the Conflict Matrix, optimized search for new setpoints using a deconfliction methodology, and creation of the resolution vector of new deconflicted setpoints. The rules-based deconfliction methodology introduced in this work is contained within the center green block and is called when a conflict has been identified and new setpoints need to be created from the set of requests received by applications.

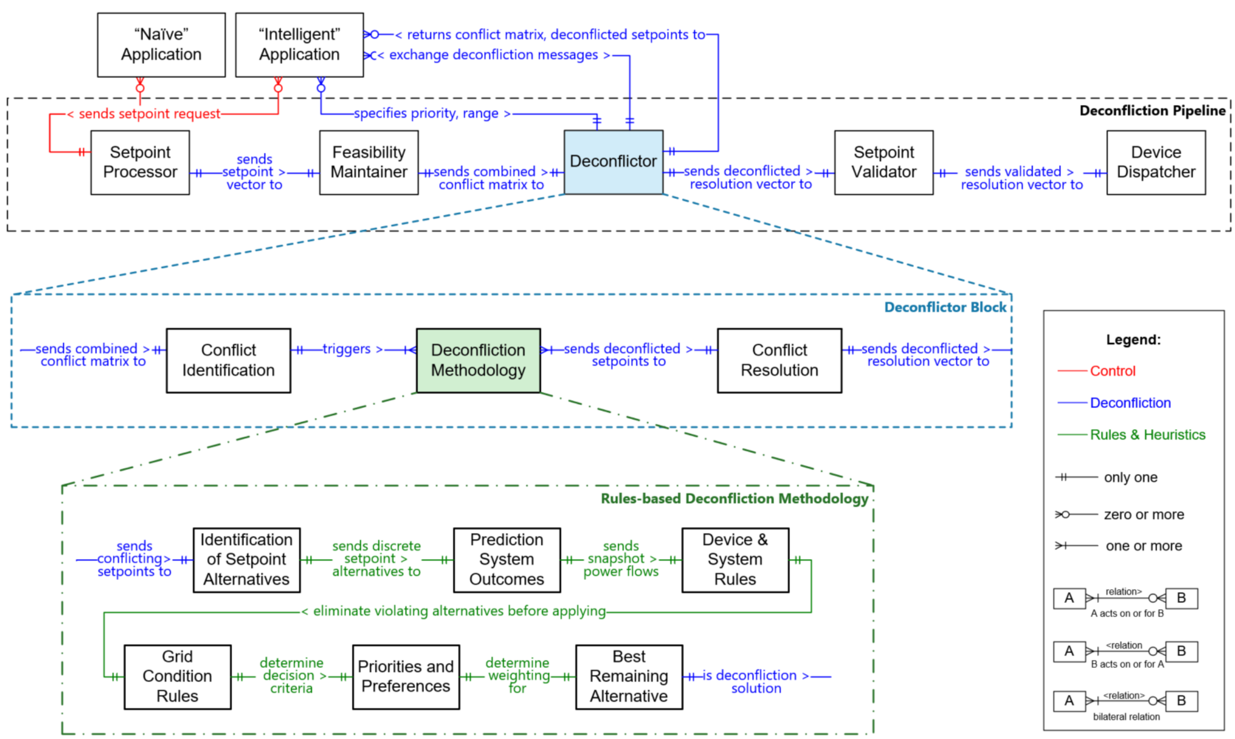

Figure 8: Location of the rules-based deconfliction methodology within the deconfliction pipeline

The rules-based deconfliction methodology, in turn, can be expanded as shown in Figure 7 and contains six sequential steps:

1) Creating discrete setpoint alternatives from the available solution space
2) Predicting outcomes for each alternative using snapshot power flow solutions
3) Eliminating non-viable alternatives using a combination of technical, economic, environmental, and social rules
4) Identifying grid operating conditions to determine the ranked order of deconfliction criteria
5) Formulating criteria weights based on the priorities and preferences of users and apps
6) Selecting the best remaining alternative that provides the highest overall combined benefit

A key consideration in the operational context of the rules-based deconfliction methodology is operational condition of the distribution grid. As described in *Distributed Rules-based Application Deconfliction, Part* 2, the grid conditions determine the specific set of device and system rules to be applied. The grid conditions are also used in the numerical multi-criteria decision-making solution to determine the rank ordering of criterion weights used to determine the local optimality of the deconfliction solution.

The Utilities Technology Council defines three conditional states in which the grid can operate [16]:

- Blue Sky conditions are optimal. The temperature is moderate, and loads are predictable and manageable. There are no weather, cyber events, physical incidents, or emergencies. Blue Sky affords utilities and generation asset owners the opportunity to engage optimization apps unimpeded, meaning that grid operations can be configured to reduce emissions, dispatch renewable resources, or participate in ancillary markets.

- Grey Sky describes conditions during which severe weather or incidents can lead to grid vulnerabilities and reliability concerns. Because Grey Sky conditions can result in outages, the grid is positioned in an advanced readiness state. Therefore, all economic optimization apps are deactivated, and physical resources are dispatched or reserved to support the distribution system or bulk power system. Grey Sky conditions are not only related to natural conditions as other threats to the health of the grid, such as a cyber event, could also warrant a Grey Sky readiness posture.

- During Black Sky conditions, a serious event has occurred, threatening widespread outages of power and/or grid communications systems. Therefore, utilities are compromised, and impacts may prevent power restoration and/or central control of grid-edge components and generation resources. Black Sky events could include disastrous storms, cyber-attack, act of war, or combination of events. During Black Sky conditions, resilience apps and assets will work collaboratively and autonomously to restore the grid, either as a whole or by piecing together islanded microgrids.

4.0 Rules-Based Deconfliction Architecture

This section derives a layered architecture for decomposing the rules-based deconfliction methodology for implementation with both centralized and distributed applications using the principles of variable structure grids [8] and the Laminar Coordination Framework [9].

4.1 Layered Decomposition of the Deconfliction Problem

The application deconfliction problem can be structurally decomposed into a series of functional layers that reflect the hierarchy of stakeholders, decision-making, and operational considerations in the deconfliction, as shown in Figure 9. The layers presented correspond to the deconfliction blocks of the Deconfliction Pipeline introduced in Section 2.6.

At the highest layer are the users of the advanced power applications and other human stakeholders directly involved in the deconfliction process (for operational roles and considerations refer to Section 2). Within the deconfliction context, the operator and dispatcher are responsible for coordinating with the shift supervisor, operations engineer, and system administrator to identify current operations priorities, configure the power applications based on system conditions, and monitor the deconfliction solution (in a manner similar to how operators currently monitor the convergence of other advanced power applications, such as contingency analysis and state estimation).

At the next level are the set of applications that are allowed to control field devices. For the purposes of this discussion, applications are divided into four categories. The first are "proprietary" applications that do not use standards-based messaging structures and attempt to send commands directly to field devices via the wide area communication network. The second are "naïve" applications that communicate on the correct IT/OT data bus using standards-based messaging structures, but repeatedly issue the same conflicting device setpoint command. Naïve applications also do not share any other information besides the requested device setpoint. The third are "intelligent" applications that share additional information to enable more accurate deconfliction, including the priority of the setpoint request, preference for setpoints of particular devices over others, and a range of acceptable setpoints that are also acceptable to the application. The final category contains distributed applications, which may be naïve or intelligent and can use a hierarchical or peer-to-peer structure for coordination and control.

The next four functional layers directly correspond the Setpoint Processor, Feasibility Maintainer, Deconflictor, Setpoint Validator, and Device Dispatcher. Multiple formulations and methodologies can be used within the deconfliction layer. This work focuses on a rules-based methodology based on discretization of setpoint ranges and elimination of infeasible alternatives based on technical, economic, environmental, and social criteria using the process block sequence shown in Figure 9.

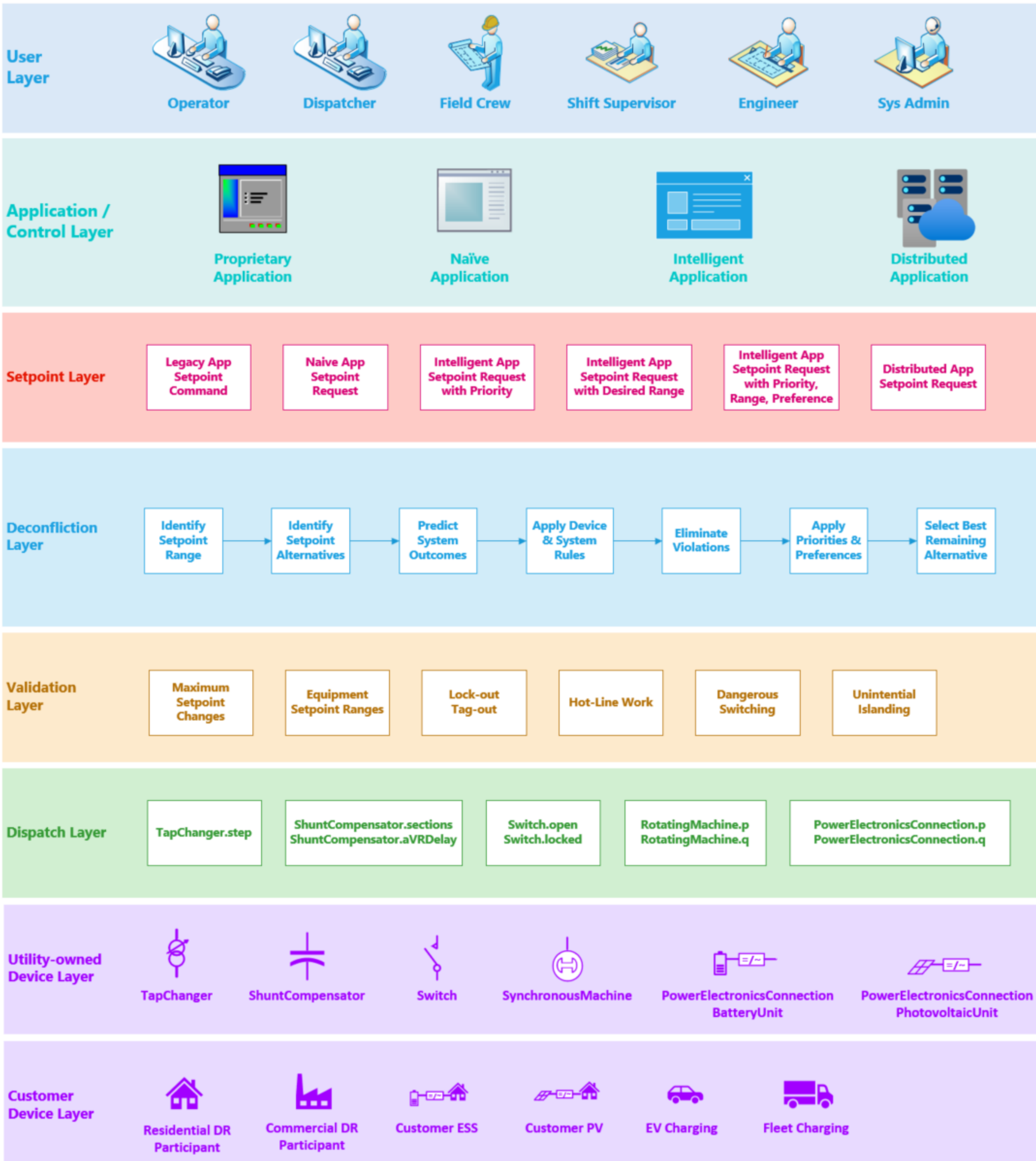

Figure 9: Decomposition of the deconfliction problem into layers of stakeholders and decision-making

Finally, the physical device layers contain utility-owned controllable devices (which can be controlled directly) and customer-owned devices (which may respond to incentive signals or be controlled as part of an opt-in program, such as demand response). The characteristics, behavior, and asset health considerations of each type of device serve as the foundation for the rules of the deconfliction methodology. A subsequent report will provide detailed descriptions of each class of device as well as variations between devices within each class (e.g. pre-programmed legacy oil-based reclosers vs modern SCADA-controlled S&C IntelliRupters).

4.2 Layered Decomposition of the Distribution Network

Simultaneously, the distribution network can be decomposed in a similar manner using the concept of variable structure grids[1] [8], which defines the set of architectural layers shown in Figure 10. At the lowest level are individual hardware components, such as reclosers, voltage regulators, underground cables, and distributed generators. These segments can then be organized into quasi-static grid segments, which only change when new hardware equipment is installed (such as construction of a new section of overhead distribution line). The grid segments (represented as part of the network planning model) are assembled into variable structure grids that adapt with topology changes due to network reconfiguration, outage response, and other real-time operations. High-level grid coordination and dispatch is performed at the next level, which assembles individual grids into a logical area network.

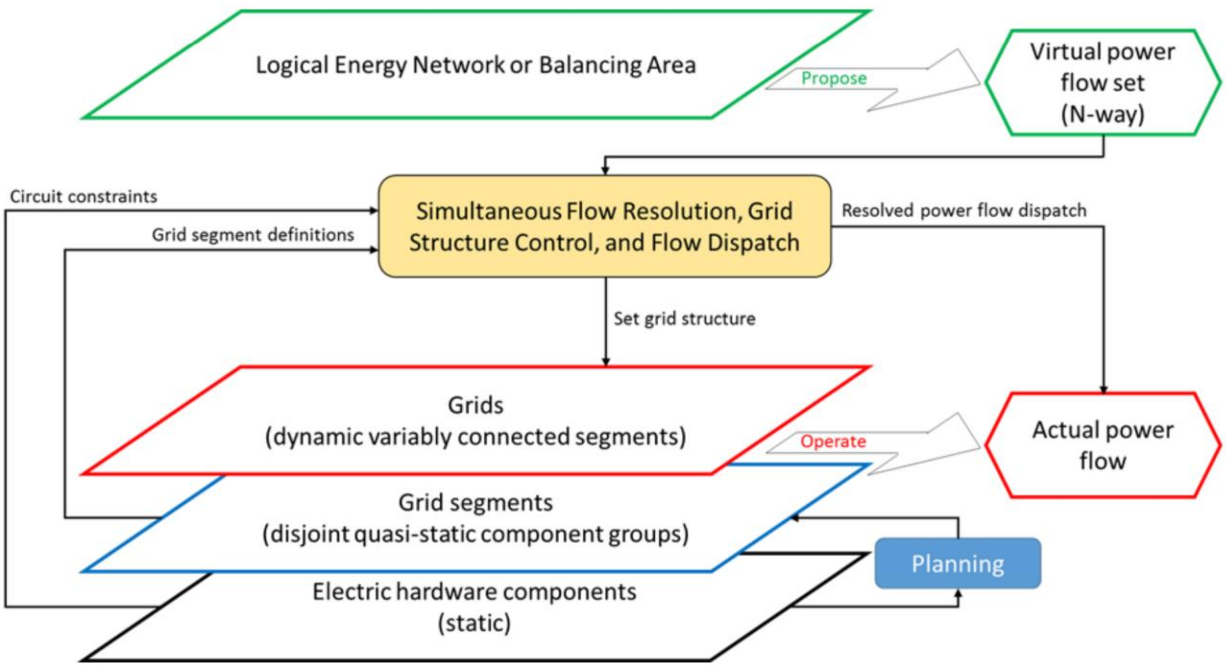

Figure 10: Decomposition of the distribution network into layers of a variable structure grid, taken from [8]

[1] A variable structure grid is defined by [8] as "grid segments (represented as subgraphs) [that] may be connected, disconnected, and/or reconnected… The as-built structure of the grid is determined by the settings of the switches and can be quite dynamic. This definition is broad and it will immediately be clear that many common practices in grid design and operation are included this definition."

To avoid the need to reconfigure the application deconflictor in response to each change in system topology or other system conditions, the deconfliction methodology introduced will focus on definition of deconfliction agents based on quasi-static component groups discussed in the next section.

4.3 Definition of Quasi-Static Grid Segments

A key concept within the Laminar Coordination Framework [9] is that of boundary deference, which states that the problem decomposition method requires discrete boundaries and interfaces. The boundaries may be defined in a variety methods, including asset ownership, service territories, communications networks, or electrical topology. For this report, topological separation of the electrical network will be used for formulation of the deconfliction methodology to provide a consistent set of boundaries and interfaces. This method of defining quasi-static grid segments will likely result in a single deconfliction agent being assigned responsibility for physical assets connected on different physical layers of the communications network (e.g. smart inverters connected via a cellular network and voltage regulators connected by leased line). The resulting challenge in delivery of control commands from the deconfliction agent to field devices is handled by the separation of the IT Data Bus from the OT Control Bus, as described in Section 2.5. Translation of the setpoint commands published on the OT Control Bus is handled automatically by an appropriate set of device protocol services that are aware of which protocol and communications network is used by each field device.

The deconfliction methodology adopts the definition of distributed topological areas defined in [17] for definition of quasi-static grid segment boundaries. This definition recognizes that utility distribution networks are defined in terms of individual feeders, which may be topologically radial or meshed. Each feeder is associated with a single substation from which it is normally energized. Energization of the feeder from alternative substations is typically defined operationally as an abnormal switching configuration. Three layers of quasi-static grid segments are defined in [17] and comprise the Feeder Area, Switch Area, and Secondary Area. The Feeder Area is defined as the topological region between the substation high-voltage transformer and the first downstream switching device. This layer contains all medium voltage (MV) substation equipment and sensors, as well as all switching devices (reclosers, sectionalizers, fuses, etc.) within the feeder. The Switch Area is defined as any section of feeder bounded by one or more switching devices. Examples include a microgrid connected by a single point of common coupling (PCC), a section of feeder between two reclosers, and a lateral downstream of a sectionalizer. Finally, the Secondary Area is bounded by the secondary service transformer and contains all low-voltage equipment and devices, including triplex lines, meters, smart inverters, and load. Examples of each type of quasi-static grid segment are illustrated for the 9500 Node Test Feeder [18] in Figure 11.

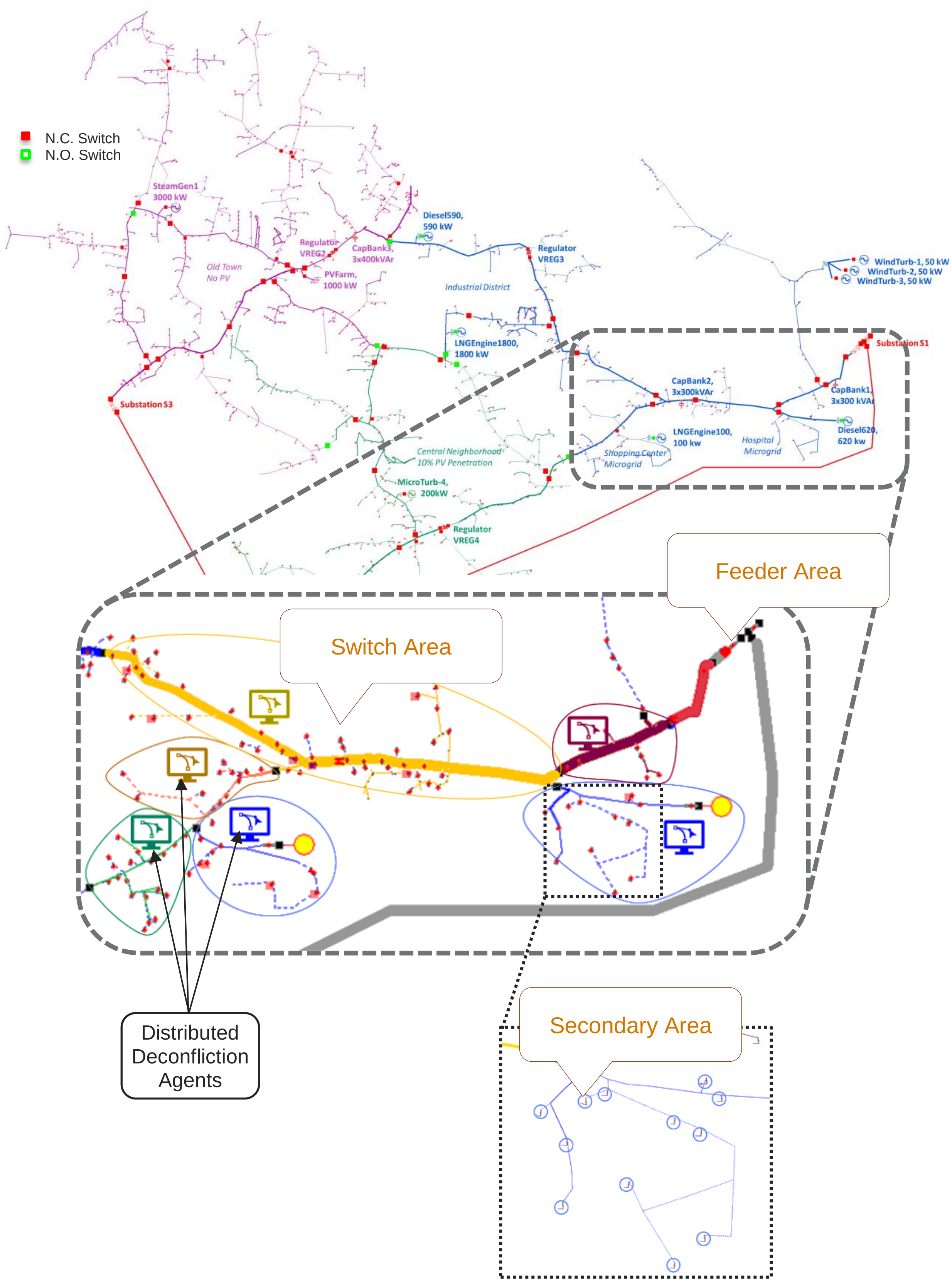

Figure 11: Decomposition of the 9500 Node Test System into individual switch areas and secondary areas

4.4 Selection of the Number of Decomposition Iterations

The rules-based deconfliction methodology introduced in this work does not apply any direct optimization approaches. Instead, it converts the solution space of possible device setpoints into a set of discrete alternatives that are eliminated based on rule-based decision criteria and then ranked from most preferred to least preferred. This approach introduces some scalability considerations that determine the optimum number of times the master deconfliction problem should be decomposed.

Discretization of the solution space results in complexity increasing linearly with the number of independent distributed topological areas but multiplicatively with the number of devices within each topological area. The scalability tradeoff can be illustrated by considering a hypothetical scenario[1] of a decarbonization application and a peak shaving application both trying to control a feeder with a voltage regulator, capacitor, and utility-scale battery located in different quasi-static grid segments of the distribution feeder. The requested setpoints of each application, shown in Table 1, result in six discrete setpoint alternatives for the voltage regulator, one for the capacitor, and nine for the battery[2].

Table 1: Discrete setpoint alternatives for two applications controlling devices in the same feeder

Device	Decarbonization Setpoint	Peak Shaving Setpoint	Setpoint Alternatives	Total Alternatives
Regulator	$+3$	$+8$	$+3, +4, +5, +6, +7, +8$	6
Capacitor	Off	Off	Off	No conflict
ESS	-400kW	$+400$kW	$-400, -300, -200, -100, 0, +100, +200, +300, +400$	9

In a fully centralized implementation (with no decomposition), these alternatives result in $6 \times 9 = 54$ pairwise setpoint combinations that need to be evaluated. However, if one topology decomposition iteration is applied, thereby placing each device in its own Switch Area, the alternatives result in $6 + 9 = 15$ setpoint combinations that evaluated by two parallel distributed deconfliction agents.

As a result, the number of decomposition iterations and number of deconfliction agents defined should be adjusted based on the number of controllable devices and their location. If the entire service territory of the electric utility contains a small number of remote-controllable devices (e.g.

[1] This scenario is not intended to be operationally realistic. It is only used for mathematical illustration of the scalability issue involved with discretization of the solution domain.
[2] Discretization of the battery setpoints into increments of 100kW is made arbitrarily. The resolution detail is decided by the user of the deconfliction methodology.

only within distribution substations), a fully centralized implementation may be acceptable. If the utility owns numerous controllable devices within each feeder, then one or two decomposition iterations may be needed to divide the distribution system into individual radial feeders and separate Switch Areas. At the highest level of complexity, if the utility has meshed low-voltage secondary networks or seeks to control thousands of rooftop solar smart inverters using multiple competing applications, then a third decomposition iteration will likely be needed with separate deconfliction agents established for each secondary area, as shown in Figure 12.

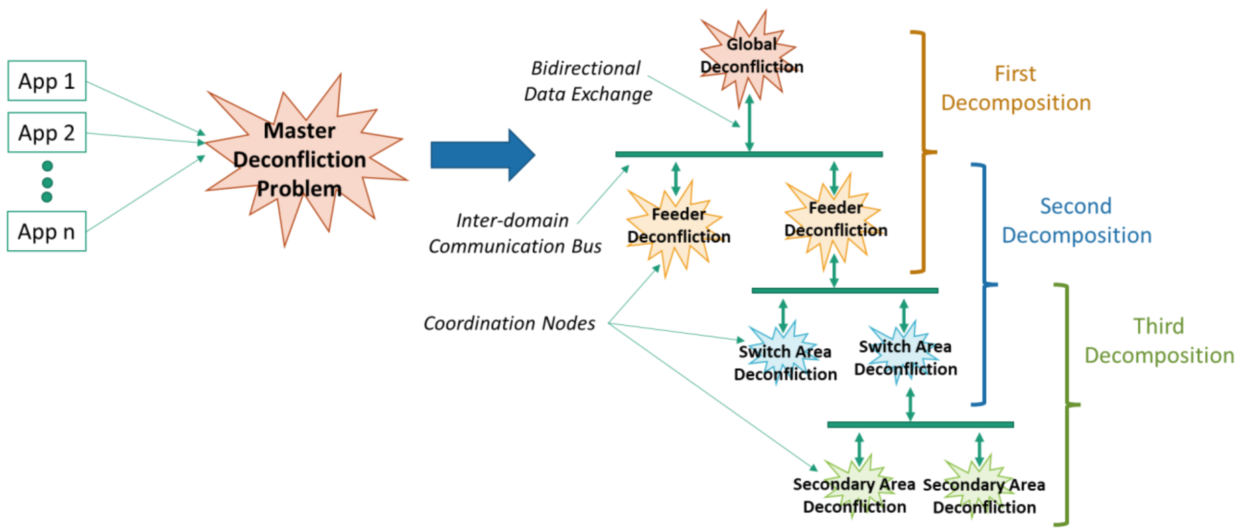

Figure 12: Decomposition iterations of the master deconfliction problem into individual subproblems limited to the controllable devices within each feeder, switch area, and secondary area.

Selection of the optimal number of decompositions of the deconfliction problem should be made based on a tradeoff analysis between computational speed and global optimality. If the deconfliction problem is decomposed too many times, the distributed decision-making process may a) have numerous software agents without controllable devices (wasting computational resources) and b) yield trivial results that do not reflect the goals of applications or the system operator. Meanwhile, if the optimization problem is not decomposed sufficiently, the solution space may contain so many devices and discrete alternatives that the prediction of system outcomes may not be able to run fast enough to keep up with evolving system conditions.

4.5 Centralized and Distributed Computational Architectures

Separation of the IT Data Bus from the OT Control Bus enables distributed deconfliction of both centralized and distributed applications, regardless of the communications and computing architecture. Decomposition of the electrical network is a software construct that may or may not be directly aligned with the location of the applications and physical devices. The laminar coordination nodes hosting each distributed deconfliction agent may be hosted in the central EMS/ADMS environment, in a secure cloud computing environment, or distributed across grid-edge devices.

In a fully centralized communications and computing system with fully centralized applications, the distributed deconfliction problem can be implemented with a parallel processing structure such that setpoints for devices each distributed topological area are delivered to the correct deconfliction agent subprocess. Each deconfliction agent is implemented as a separate software service with its own set of processor cores, which it uses to create predictive estimates of the system outcomes, eliminate alternatives in violation of system rules, and identify the most preferred remaining alternative (as illustrated previously in the deconfliction layer of Figure 8). The deconflicted setpoint solutions are then validated are passed to the correct centralized device protocol services and then broadcast over the communications network(s) to the appropriate devices.

In a fully distributed communications and computing architecture with fully distributed and/or decentralized applications, the deconfliction problem is handled locally by grid-edge computing resources. Each grid-edge deconfliction agent is aware of the controllable devices within its distributed topological area and solves a local problem for its own set of devices. Depending on the computing architecture, the deconfliction agents can either be fully decentralized or coordinate with each other in a hierarchical and/or peer-to-peer manner.

In a hybrid environment, with a combination of centralized applications hosted in the utility control room and distributed applications hosted on grid-edge resources, a distributed computing approach may be applied. The distributed deconfliction agent for each topological area would be hosted on the same grid-edge network as each distributed application. Setpoints from the centralized applications and distributed applications for the specific topological area would be delivered to or intercepted by the distributed deconfliction agent, which would then solve its local decision-making problem in a manner similar to that described for the fully distributed solution.

References

[1] A. P. Reiman, S. Poudel, G. Mukherjee, A. A. Anderson, O. Vasios, T. E. Slay, G. D. Black, A. Dubey and J. P. Ogle, "App deconfliction: Orchestrating distributed, multi-agent, multi-objective operations for power systems," *submitted to IEEE Access,* 2023.

[2] M. Vadari, Electric system operations - Evolving to the Modern Grid, 2nd Ed., Boston: Artech House, 2020.

[3] S. S. Amiri, M. Rahmani and J. D. McDonald, "An updated review on distribution management systems within a smart grid structure," in *IEEE Smart Grid Conference*, 2021.

[4] A. Dubey, A. Bose, M. Liu and L. N. Ochoa, "Paving the way for advanced distribution management systems applications: Making the most of models and data," *IEEE Power & Energy Magazine,* vol. 18, no. 1, pp. 63 - 75, 2020.

[5] A. A. Anderson, E. G. Stephan and T. E. McDermott, "Enabling Data Exchange and Data Integration with the Common Information Model," Pacific Northwest National Laboratory, PNNL-32679, Richland, WA, 2022.

[6] J. D. Taft and A. Becker-Dippman, "Grid Architecture," Pacific Northwest National Laboratory, PNL-24044, 2015.

[7] R. B. Melton, S. Pal and J. Taft, "Problem domain reference model - GMLC 1.2.1," Pacific Northwest Na, Richland, WA, 2020.

[8] J. D. Taft, "Architecture framework for variable structure grids (PNNL-32515)," Pacific Northwest National Laboratory, Richland, WA, 2021.

[9] J. D. Taft, "Comparative architecture analysis: Using laminar structure to unify multiple grid architectures," Pacific Northwest National Laboratory, PNNL-26089, 2016.

[10] J. D. Taft, "Grid Architecture 2," Pacific Northwest National Laboratory, PNL-24044-2, 2016.

[11] S. Vadari, "Evolving architectures and considerations to address distributed energy resources and non-wired alternatives," Pacific Northwest National Laboratory, PNNL-31751, 2021.

[12] International Electrotechnical Commission, "IEC 61968-1:2020 Application integration at electric utilities - System interfaces for distribution management - Part 1: Interface architecture and general recommendations," Apr 2020. [Online]. Available: https://webstore.iec.ch/publication/32542. [Accessed Dec 2022].

[13] J. D. Taft, "Architectural basis for highly distributed transactive power grids: Frameworks, networks, and grid codes (PNNL-25840)," Pacific Northwest National Laboratory, Richland, WA, 2016.

[14] A. Kossiakoff, W. N. Sweet, S. J. Seymour and S. M. Biemer, Systems engineering: Principles and practice, 2nd Ed., Hoboken, NJ: Wiley, 2011.

[15] J. D. Taft and J. P. Ogle, "Grid architecture guidance specification for FAST-DERMS (PNNL-31172)," Pacific Northwest National Laboratory, Richland, WA, 2021.

[16] Utilities Technology Council, "Glossary of Terms," October 2018. [Online]. Available: https://utc.org/wp-content/uploads/2018/10/Definitions_Final-Version_October-2018.pdf. [Accessed March 2023].

[17] A. A. Anderson, R. Podmore, P. Sharma, A. P. Reiman, R. A. Jinsiwale, C. H. Allwardt and G. D. Black, "Distributed application architecture and LinkNet topology processor for

distribution networks using the Common Information Model," *IEEE Access,* vol. 10, pp. 120765 - 120780, 2022.

[18] A. A. Anderson, S. Vadari, J. Barr, S. Poudel, A. Dubey, T. McDermott and R. Podmore, "Introducing the 9500 Node Test System to support advanced power applications," Pacific Northwest National Laboratory, PNNL-33471, Richland, WA, 2022.

[19] Vermont Electric Cooperative, "Reliability," [Online]. Available: https://vermontelectric.coop/electric-system/reliability. [Accessed 15 Mar 2022].

**Pacific Northwest
National Laboratory**

902 Battelle Boulevard
P.O. Box 999
Richland, WA 99354

1-888-375-PNNL (7665)

www.pnnl.gov