[Unit 1](#_bookmark0)

[Medical Cyber- Physical Systems](#_bookmark0)1

* Medical cyber-physical systems (MCPS) are life-critical, context-aware, networked systems of medical devices that are collectively involved in treating a patient.
* These systems are increasingly used in hospitals to provide high-quality continuous care for patients in complex clinical scenarios.
* The need to design complex MCPS that are both safe and effective has presented numerous challenges, inclulding achieving high levels of assurance in system software, interoperability, context- aware decision support, autonomy, security and privacy, and certification.

3

#### [Introduction and Motivation](#_bookmark0)

* The two most significant transformations in the field of medical devices in recent times are the high degree of reliance on software-defined functionality and the wide availability of network connectivity.

*Some of the research directions being taken to address the various challenges involved in building MCPS:*

* + - ***Stand-alone device****:* A model-based high-assurance software development scheme is described for stand-alone medical devices such as patient-controlled analgesia (PCA) pumps and **pacemakers.**
    - ***Device interconnection:***A medical device interoperability framework is presented for describing, instantiating, and validating clinical **interaction scenarios.**
    - ***Adding intelligence:***A smart alarm system is presented that takes vital signs data from various interacting devices to **inform caregivers of potential patient emergencies** and non-operational issues about the devices.
    - ***Automated actuation/delivery:***A model-based closed-loop care delivery system is presented, which can **autonomously deliver care** to the patients based on the current state of the patient.
    - ***Assurance cases:***The use of assurance cases is described for organizing collections of claims, arguments, and **evidence** to establish the safety of a medical device system.

#### [System Description and Operational Scenarios](#_bookmark0)

**MCPS are safety-critical, smart systems of interconnected medical devices that are collectively involved in treating a patient within a specific *clinical scenario*.**

MCPS alter this view by introducing additional computational entities that aid the caregiver in controlling the “plant.” Figure below presents a conceptual overview of MCPS.

Devices used in MCPS can be categorized into **two large** groups based on their primary functionality:

* ***Monitoring devices***, such as bedside heart rate and oxygen level monitors and sensors, which provide different kinds of clinic- relevant information about patients
* ***Delivery devices*,** such as infusion pumps and ventilators, which actuate therapy that is capable of changing the patient’s physiologi- cal state



I *K e (τ)dτ*

*t*

*i* 

*0*

Monitoring Medical Devices

Smart Controller

Smart Alarm

Caregiver

Patient

Setpoint   Error



 Process Output

Decision Support

Administrative Support

D *Kd de*(*t*) *dt*

P *Kpe*(t)

Treatment Delivery Medical Devices

**Figure :** *A conceptual overview of medical cyber-physical systems*

* In MCPS, interconnected monitoring devices can feed collected data to decision support or administrative support entities, each of which serves a different purpose.
* Alternatively, the decision support entities might utilize a smart controller to analyze the data received from the monitoring devices, estimate the state of the patient’s health, and automatically initiate treatment (e.g., drug infusion) by issuing commands to delivery devices, thereby closing the loop.
* One of the more effective strategies to do so is to use model-based development methods, which can ensure device safety by enabling medical device verification.

##### [Virtual Medical Devices](#_bookmark0)

* Due to high complexity of MCPS, we sbould develop a description of the MCPS workflow and then enforce it on physical devices.
* MCPS workflow can be described in terms of the

1)number and types of devices involved,

2) their mutual interconnections, and

3) the clinical supervisory algorithm needed for coordination and analysis of data collected by the system.

* Such a description defines *virtual medical device* (VMD). VMDs are used by a *VMD app* and instantiated during the setup of actual medical devices—that is, as part of a *virtual medical device instance*.
* **The principal task** of the VMD app, therefore, is to find the medical devices in a VMD instance (which may be quite large), establish network connections between them, and install the clinical algorithm into the supervisor module of the middleware for managing the interactions of the clinical workflow and the reasoning about the data produced.
* Basically, when the VMD app is started, the supervisor reads the VMD app specification and tries to couple all involved devices according to the specification. Once the workflow has run its course,
* VMD app can perform the necessary cleanup to allow another workflow to be specified using a different combination of medical devices in the VMD instance.

##### [Clinical Scenarios](#_bookmark0)

* Each VMD supports a specific clinical scenario with a detailed descrip- tion of how devices and clinical staff work together in a clinical situa- tion or event.
* Two such scenarios: one for X ray and ventilator coordination and another for a patient-controlled analgesia (PCA) safety interlock system.
* Consider the scenario where X-ray images are often taken during surgical procedures. If the surgery is being performed under general anesthesia, the patient breathes with the help of a ventilator during the procedure. Because the patient on ventilator cannot hold his or her breath to let the X-ray image be taken without the blur caused by moving lungs, the ventilator has to be paused and later restarted. In some unfortunate cases, the ventilator was not restarted, leading to the death of the patient.
* A safer way, is to let the ventilator transmit its internal state to the X-ray machine. There typically is enough time to take an X-ray image at the end of the breathing cycle, between the time when the patient has finished exhaling and the time he or she starts the next inhalation.
* **Another clinical** scenario is patient-controlled analgesia. PCA infusion pumps are commonly used to deliver opioids for pain management— for instance, after surgery.
* Patients have very different reactions to the medications and require distinct dosages and delivery schedules.
* PCA pumps allow patients to press a button to request a dose when they decide they want it, rather than using a dosing schedule fixed by a caregiver.
* A major problem with opioid medications in general is that an excessive dose can cause respiratory failure.
* A properly programmed PCA system should prevent an overdose by limiting how many doses PCA infusion pumps are currently associated with a large number of adverse events, and existing safeguards such as drug libraries and programmable limits are not adequate to address all the scenarios seen in clinical practice.

#### [Key Design Drivers and Quality Attributes](#_bookmark0)

##### [Trends](#_bookmark0)

Four trends in MCPS are critical in the evolution of the field: **software** as the main driver of new features, **device interconnection**, **closed loops** that automatically adjust to physiological response, and a new focus on **continuous monitoring** and care. The following subsections discuss each of these trends.

###### New Software-Enabled Functionality

* Introduction of new functionality is largely driven by the new possibilities that software-based development of medical device systems is offering.
* A prime example of the new functionality is seen in the area of robotic surgery, which requires real-time processing of high-resolution images and haptic feedback.
  + Another example is proton therapy treatment. One of the most technology-intensive medical procedures, it requires one of the largest- scale medical device systems. To deliver its precise doses of radiation to patients with cancer, the treatment requires precise guiding of a proton beam from a cyclotron to patients, but must be able to adapt to even minor shifts in the patient’s position.
  + Control of proton beams is subject to very tight timing constraints, with much less tolerance than for most medical devices.
  + To further complicate the problem, the same beam is applied to multiple locations in the patient’s body and needs to be switched from location to location, opening up the possibility of interference between beam scheduling and application.
  + In addition to controlling the proton beam, a highly critical function of software in a proton treatment sys tem is real-time image processing to determine the precise position of the patient and detect any patient movement.
  + *I*n simpler devices, such as pacemakers and infusion pumps, more and more software-based features are being added, making their device software more complex and error prone.

###### Increased Connectivity of Medical Devices

* + In addition to relying on software to a greater extent, medical devices are increasingly being equipped with network interfaces.
  + In essence, interconnected medical devices form a distributed medical device system of a larger scale and complexity that must be properly designed and validated to ensure effectiveness and patient safety.
  + The networking capabilities of most medical devices today are lim- ited in functionality and tend to rely on proprietary communication protocols offered by major vendors.
  + There is, however, a growing realization among clinical professionals that open interoperability between different medical devices will lead to improved patient safety and new treatment procedures.
  + The Medical Device Plug-and-Play (MD PnP) Interoperability initiative is a relatively recent effort that aims to provide an open standards framework for safe and flexible interconnectivity of medical devices, with the ultimate goal of improving patient safety and health care efficiency.
  + In addition to developing interoperability standards, the MD PnP initiative collects and demonstrates clinical scenarios in which interoperability leads to improvement over the existing practice.

###### Physiological Closed-Loop Systems

* + - * Traditionally, most clinical scenarios have a caregiver—and often more than one—controlling the process.
      * There is a concern in the medical community that reliance on humans being in the loop may compromise patient safety.
* Caregivers, who are often overworked and operate under severe time pressures, may miss a critical warning sign. Nurses, for example, typically care for multiple patients at a time and can become distracted.
  + - * Using an automatic controller to provide continuous monitoring of the patient state and handling of routine situations would relieve some of the pressure on the caregiver and might potentially improve patient care and safety.
      * Although the computer will probably never replace the caregiver completely, it can significantly reduce the workload, calling the caregiver’s attention only when something out of the ordinary happens.
      * Scenarios based on physiological closed-loop control have been used in the medical device industry for some time.

###### Continuous Monitoring and Care

* Mobile monitoring and home monitoring of vital signs and physical activities allow health to be assessed remotely at all times.
* Sophisticated technologies such as body sensor networks to measure training effectiveness and athletic performance based on physiological data such as heart rate, breathing rate, blood sugar level, stress level, and skin temperature are becoming more popular.
* However, most of the current systems operate in store- and-forward mode, with no real-time diagnostic capability.
* Physiological closed-loop technology will allow diagnostic evaluation of vital signs in real time and make constant care possible.

##### [Quality Attributes and Challenges of the MCPS Domain](#_bookmark0)

Building MCPS applications requires ensuring the following quality attributes, which in turn pose significant challenges:

* *Safety:* High-confidence software development is critical to ensure the safety and effectiveness of MCPS. We advocate the use of model-based development and analysis as a means of ensuring the safety of MCPS.
* *Interoperability:* It is essential to ensure that the MCPS built from interoperable medical devices are safe, effective, and secure, and can eventually be certified as such.
  + *Context-awareness:* Integration of patient information from multiple sources can provide a better understanding of the state of the patient’s health, with the combined data then being used to enable early detection of ailments and generate effective alarms in the event of emergency.
  + *Autonomy:* The computational intelligence that MCPS possess can be applied to increase the autonomy of the system by enabling actuation of therapies based on the patient’s current health state.

Safety analysis of autonomous decisions in the resulting closed- loop system is a major challenge, primarily due to the complexity and variability of human physiology.

* + *Security and privacy:* Medical data collected and managed by MCPS are very sensitive. Unauthorized access or tampering with this information can have severe consequences to the patient in the form of privacy loss, discrimination, abuse, and physical harm. increasing the vulnerability of the system to security and privacy violations.
  + *Certification:* The complex and safety-critical nature of MCPS requires a cost-effective way to demonstrate medical device software dependability. The notion of assurance cases holds the promise of providing an objective, evidence-based approach to software certification.

##### [High Confidence - Development of MCPS](#_bookmark0)

* The extreme market pressures faced by the medical devices industry has forced many companies to reduce their development cycles as much as possible.
* The challenge is to find a development process that will deliver a high degree of safety assurance under these conditions.
* Model-based development can be a significant part of such a develop- ment process.
* The case study discussed in this section illustrates the steps of the high-assurance development process using a simple medical device.
* The choice of modeling, verification, and code generation technologies depends on factors such as complexity and criticality level of the appli cation.

###### Mitigation of Hazards

* Most of the new functionality in medical devices is software based, and many functions traditionally implemented in hardware—including safety interlocks—are now being relegated to software.
* Thus, high confidence software development is very important for the safety and effectiveness of MCPS.
* In the below figure the process starts with the identification of the desired functionality and the hazards associated with the system’s operation.
* The chosen functionality yields the system functional requirements, while hazard mitigation strategies yield the system safety requirements.



Hazard analysis and mitigation

Functional requirements elicitation

**Verification**

Model-based development

**Code**

**generation**

**Validation**

Behavioral model

Safety properties

**Figure :** *High-assurance development process for embedded software*

* The functional requirements are used to build detailed behavioral models of the software modules, while the safety requirements are turned into properties that these models should satisfy.
* Models and their desired properties are the inputs to the model-based software development, which consists of verification, code generation, and validation phases.
* Model-based development has emerged as a means of raising the level of assurance in software systems.
* In this approach, developers start with declarative models of the system and perform a rigorous model verification with respect to safety and functional requirements;
* It uses systematic code generation techniques to derive code that preserves the verified properties of the model. Such a development process allows the developers to detect problems with the design and fix them at the model level, early in the design cycle, when changes are easier and cheaper to make.
* More importantly, it holds the promise of improving the safety of the system through verification.
* The use of formal modeling facilitates making mathematically sound conclusions about the models and generating code from them.

###### Challenges of Model-Driven Development of MCPS

* The first challenge is choosing the right level of abstraction for the modeling effort.
* A highly abstract model makes the verification step relatively easy to perform, but a model that is too abstract is difficult to use in the code generation pro cess, since too many implementation decisions have to be guessed by the code generator.
* Conversely, a very detailed model makes code generation relatively straight forward, but pushes the limits of the currently available verification tools.
* From the modeling and verification perspective, there are several reasons to separate the platform-independent aspects from the platform- dependent aspects.
* For code generation, one may need to specify the details of how the device retrieves data from sensors.
* A sampling-based mechanism with a particular sampling interval will yield a very different generated code compared to an interrupt-based mechanism.
* Abstracting from a particular platform allows us to use the model across different target platforms.
* Different platforms may have different kinds of sensors that supply the same value.
* For example, consider an empty-reservoir alarm, such as that implemented on many infusion pumps. Some pumps may not have a physical sensor for that purpose and simply estimate the remaining amount of medication based on the infusion rate and elapsed time. Other pumps may have a sensor based on syringe position or pressure in the tube.
* There is a semantic gap between the model and the implementation. A system is modeled using the formal semantics pro vided by the chosen modeling language.
* The following case study concentrates on the development of a PCA infusion pump system and considers several approaches to

address these challenges.

###### Case Study: PCA Infusion Pumps

* A PCA infusion pump primarily delivers pain relievers, and is equipped with a feature that allows for additional limited delivery of medication, called a bolus, upon patient demand.
* This type of infusion pump is widely used for pain control of postoperative patients.
* If the pump overdoses opioid drugs, however, the patient can be at risk of respiratory depression and death. Therefore, these medical devices are subject to stringent safety requirements that aim to prevent overdose.

###### The Generic PCA Project - CASE STUDY

* The Generic PCA (GPCA) project, a joint effort between the PRECISE Center at the University of Pennsylvania and researchers at the FDA, aims to develop a series of publicly available artifacts that can be used as guidance for manufacturers of PCA infusion pumps.
* In the first phase of the project, a collection of documents has been developed, including a hazard analysis report, a set of safety requirements, and a reference model of PCA infusion pump systems.
* Based on these documents, companies can develop PCA infusion pump controller software following a model-driven implementation.
* In the case study, software for the PCA pump controller is developed by using the model-driven implementation approach starting from the reference model and the safety requirements.
* The case study included the construction of an assurance case—a structured argument based on the evidence collected during the development process, which aims to convince evaluators that the GPCA-reference implementation complies with its safety requirements.

###### Modeling

* The reference model of the GPCA pump implemented in Simulink/ Stateflow is used as the source of functional requirements and converted to UPPAAL via a manual but systematic translation process. The model structure follows the overall architecture of the reference model, which is shown in Figure below.
* The software is organized into two state machines: the state controller and the alarm-detecting component.
* Both state machines interact with sensors and actuators on the pump platform.

Test sequences

GPCA safety requirements

Manual translation

Manual translation

UPPAAL model

External channels, clock source

Manual implementation

Code synthesis (TIMES tool)

Formal verification

Platform-dependent glue code

Simulation

Verification result

Code-interfacing compilation

Model trace

Implementation trace

Validation

Test sequences

Executable image for the target platform

UPPAAL queries

GPCA model (Simulink/Stateflow)

Validation result

Platform-independent code

**Figure 1.3:** *The model-driven development for the GPCA prototype*

User Interface

**Figure :** *The system architecture of the GPCA model*

GPCA Model

Current Failure Condition

Alarm/Warning Notification

Pump Ready/Not Ready

Infusion In Progress Clear Alarm

Drug Library Information

Bolus Status Basal Infusion Status

Current State

Alarm/Warning Notification

System Model

Failure/Anomaly Flags

User Action Vector

User Data Input

Infusion Configuration Routine

Check Drug Routine

Infusion Session Submachine

POST

State Controller

Alarm Detecting Component

Drug Library Information

Bolus Status

Basal Infusion Status

Infusion Control Signals

Infusion Program

The state machines are organized as a set of modes, with each mode captured as a separate submachine. In particular, the state controller contains four modes:

* Power-on self-test (POST) mode is the initial mode that checks sys- tem components on start-up.
  + The check-drug mode represents a series of checks that the care- giver performs to validate the drug loaded into the pump.
  + The infusion configuration mode represents interactions with the caregiver to configure infusion parameters such as infusion rate and volume to be infused (VTBI) and validate them against the lim- its encoded in the drug library.
  + The infusion session is where the pump controls delivery of the drug according to the configuration and the patient’s bolus requests.

###### Model Verification

* Representative requirements are “No normal bolus doses shall be administered when the pump is alarming” and “The pump shall issue an alert if paused for more than *t* minutes.”
* Before verification can be performed, requirements need to be for- malized as properties to be checked.
* We can categorize the requirements according to their precision and level of abstraction:

*Category A:* Requirements that are detailed enough to be formalized and verified against the model

*Category B:* Requirements that are beyond the scope of the model

*Category C:* Requirements that are too imprecise to be formalized

* + Only requirements in Category A can be readily used in verifica- tion. Just 20 out of the 97 GPCA requirements fell into this category.
  + Most of the requirements in Category B concern the functional aspects of the system that are abstracted away at the modeling level. For example, consider the requirement “If the suspend occurs due to a fault condition, the pump shall be stopped immediately without com- pleting the current pump stroke.”
  + There is another requirement to com- plete the current stroke under other kinds of alarms.
  + Thus, the motor needs to be stopped in different ways in different circumstances.
  + These requirements fall into Category B, since the model does not detail the behavior of the pump stroke.
  + Handling of properties in this category can be done in several ways.
  + A better approach is to match the level of detail by further decomposing the requirements.
  + At the platform-independent level, we might check that the system performs two different stop actions in response to different alarm conditions (which would be a Category A requirement).
  + Then, at the platform-specific level, we might check that one stop action corresponds to immediate stopping of the motor, while the other stop action lets the motor complete the current stroke.

###### Code Generation and System Integration

* + Once the model is verified, a code generation tool is used to produce the code in a property-reserving manner.
  + An example of such a tool is TIMES [Amnell03] for UPPAAL timed automata.
  + Input and output actions (e.g., a bolus request by a patient or triggering of the occlusion alarm from the pump hardware) are abstracted as instantaneous transitions subject to input/output syn- chronization with their environment.
  + On a particular platform, the underlying operating system schedules the interactions, thereby affecting the timing of their execution.
  + The platform integration is then performed by verifying time-safety—that is, checking whether the platform-independent code can be scheduled on the particular platform.
  + Another approach is to systematically generate an I/O interface that helps the platform-independent and dependent code to be integrated in a traceable manner.
  + From a code generation perspective, proposed a way to generate code for a given composite block of the model independently from context and using minimal information about the internals of the block.

###### Validation of the Implementation

* Unless the operation of an actual platform is completely formalized, inevitably some assumptions will be made during the verification and code generation phases that cannot be formally guaranteed.
* The validation phase is meant to check that these assumptions do not break the behavior of the implementation.
* In the case study, a test harness systematically exercises the code using test cases derived from the model.
* A rich literature on model-based test generation exists; for a survey of the area.
* The goal of such testing-based validation is to systematically detect deviations of the system behavior from that of the verified model.

##### [On-Demand Medical Devices and Assured Safety](#_bookmark0)

The final system is assembled by the user instead of the manufacturer.

Research into the safety assessment of these systems is actively under way.

The success and safety of these systems will depend not only on new engineering techniques, but also on new approaches to regulation and the willingness of industry members to adopt appropriate interoperability standards.

###### Device Coordination

* To provide complex therapy, caregivers (i.e., physi- cians and nurses) must coordinate the activities of the various medical devices manually. This is burdensome for the caregiver, and prone to errors and accidents.
* An example is trachea or larynx surgery performed with a laser scalpel. In this type of surgery, the patient is placed under general anesthesia while the surgeon makes cuts on the throat using a high-intensity laser. Because the patient is under anesthesia, his or her breathing is supported by an anesthesia ventilator that supplies a high concentration of oxygen to the patient. This situation presents a serious hazard:
* If the surgeon accidentally cuts into the breathing tube using the laser, the increased concentration of oxygen can lead to rapid combustion, burning the patient from the inside out.
* To mitigate this hazard, the surgeon and the anesthesiologist must be in constant communication: When the surgeon needs to cut, he or she signals the anesthesiologist, who reduces or stops the oxygen being supplied to the patient.
* If the patient’s oxygenation level drops too low, the anesthesiologist signals the surgeon to stop cutting so oxygen can be supplied again.
* Many other clinical scenarios might benefit from this kind of automated medical device coordination.
* These scenarios involve either *device synchronization*, *data fusion*, or *closed-loop control*.
* Finally, closed-loop control of therapy can be achieved by collecting data from devices that sense the patient’s physiological state and then using those data to control actuators such as infusion pumps .

###### Definition: Virtual Medical Devices

* A collection of devices working in unison to implement a given clinical scenario is, in essence, a new medical device.
* Such collections have been referred to as virtual medical devices (VMDs) because no single manufacturer is producing this device and delivering it fully formed to the clinician.
* A VMD does not exist until assembled at the patient’s bedside.
* A VMD instance is created each time the clinician assembles a particular set of devices for the VMD and connects them together.

###### Standards and Regulations

* + Several existing standards are designed to enable medical device inter- connectivity and interoperability.
  + While these standards enable medical devices to exchange and interpret data, they do not adequately address more complex interactions between medical devices, such as the inter-device coordination and control needed with the laser scalpel and ventilator combination.
  + VMD poses a major fundamental question of **safety in systems** that are assembled by their users.
  + Traditionally, most safety-critical cyber-physical systems, such as aircraft, nuclear power plants, and medical devices, are evaluated for safety by regulators before they can be used.
  + Virtual medical devices are constructed at bedside, based on the needs of an individual patient and from available devices.
  + This means that a caregiver may instantiate a VMD from a combination of medical devices (i.e., varying in terms of make, model, or feature set) that have never been combined into an integrated system for that particular clinical scenario.
  + Finally, “on-demand” instantiation of the VMD confounds the regulatory pathways for medical devices that are currently available.

###### Case Studies

* The subject of safety assessment of on-demand medical systems has been the focus of a number of research projects.
* These projects have explored different aspects of on-demand medical systems, their safety, and possible mechanisms for regulatory oversight.
* The Medical Device Plug-and-Play project articulated the need for on-demand medical systems, documented specific clinical scenarios that would benefit, and developed the Integrated Clinical Environment (ICE) architecture.
* In such an architecture, each medical system would be composed out of a variety of components (clinical applications, a medical application platform, and medical devices), which would be regulated, certified, and then obtained by the healthcare organization separately.

###### Integrated Clinical Environment

* The Figure shows the primary components of the integrated clinical environment (ICE) architecture.
* This case study summarizes the intended functionality and goals for each of these components.

**Supervisor**

ICE App Code Language / Virtual Machine

App An

...

App A2

App A1

Physical Device

Native

EI-Compliant Physical Device

EI

Adapter

ICE Equipment Interface (EI) I3

ICE Equipment Interface (EI) I1

ICE EI Interface Description Language

**Network Controller (NC)**

|  |
| --- |
| EI  Adapter |
|  |
| Physical Device |

**Figure** *ICE architecture*

Nevertheless, the roles of each of the components in the architecture imply certain informal requirements:

* + *Apps:* Applications are software programs that provide the coordi- nation algorithm for a specific clinical scenario (i.e., smart alarms, closed-loop control of devices). In addition to executable code, these applications contain device requirements declarations—that is, a description of the medical devices they need to operate cor- rectly.
  + *Devices:* Symmetrical to the applications, medical devices used in the ICE architecture would implement an interoperability standard and carry a self-descriptive model, known as a capabilities specifi- cation.
  + *Supervisor:* The supervisor provides a secure isolation kernel and virtual machine (VM) execution environment for clinical applica- tions. It would be responsible for ensuring that apps are partitioned in both data and time from each other.
  + *Network controller:* The network controller is the primary conduit for physiologic signal data streams and device control messages. The network controller would be responsible for maintaining a list of connected devices and ensuring proper quality of service guarantees in terms of time and data partitioning of data streams, as well as security services for device authentication and data encryption.
  + *ICE interface description language:* The description language is the primary mechanism for ICE-compliant devices to export their capa- bilities to the network controller. These capabilities may include which sensors and actuators are present on the device, and which command set it supports.

###### Medical Device Coordination Framework

The Medical Device Coordination Framework (MDCF) is an open-source project that aims to provide a software implementation of a medical application platform that conforms to the ICE standard.

The modular framework is envisioned as enabling researchers to rapidly prototype systems and explore implementation and engineering issues associated with on-demand medical systems.

The MDCF is implemented as a collection of services that work together to provide some of the capabilities required by ICE as essential for a medical application platform.

The functionality of these services also may be decomposed along the architectural boundaries defined in the ICE architecture, that is, the MDCF consists of **network controller services, supervisor services, and a global resource management service.**

Network controller services are as follows:

* *Message bus:* Abstracts the low-level networking implementation (e.g., TCP/IP) and provides a publish/subscribe messaging service. All communication between medical devices and the MDCF occurs via the message bus, including protocol control messages, exchanges of patient physiologic data, and commands sent from apps to devices. The message bus also provides basic real-time guarantees (e.g., bounded end-to-end message transmission delays) that apps can take as assumptions. Additionally, the message bus supports various fine-grained message and stream access control and isolation

Supervisor

App manager

Resource service

Network controller

Message bus

Data logger

Medical device n

Medical device 1

Key

Pub/Sub interface Private API interface

Device database

Device manager

Admin service

Clinician service

App database

...

**Figure 1.6:** *MDCF services decomposed along ICE architectural boundaries*

policies. The message bus encodes messages using XML.

* + *Device manager:* Maintains a registry of all medical devices cur- rently connected with the MDCF. The device manager implements the server side of the MDCF device connection protocol (medical devices implement the client side) and tracks the connectivity of those devices, notifying the appropriate apps if a device goes offline unexpectedly. The device manager validates the trustworthiness of any connecting device by determining whether the connecting device has a valid certificate.
  + *Device database:* Maintains a list of all specific medical devices that the healthcare provider’s bioengineering staff has approved for use. In particular, the database lists each allowed device’s unique identifier (e.g., an Ethernet MAC address), the manufacturer of the device, and any security keys or certificates that the device man- ager will use to authenticate connecting devices against.
  + *Data logger:* Taps into the flows of messages moving across the mes- sage bus and selectively logs them. Because the message bus carries every message in the system, the logger can be configured to record any message or event that prop- agates through the MDCF. Logs must be tamper resistant and tam- per evident; access to logs must itself be logged, and be physically and electronically controlled by a security policy.

Supervisor services are as follows:

* + *Application manager:* Provides a virtual machine for apps to execute in. In addition to simply executing program code, the application manager checks that the MDCF can guarantee the app’s require- ments at runtime and provides resource and data isolation, as well as access control and other security services. If the app requires a certain medical device, communications latency, or response time from app tasks, but the MDCF cannot currently make those guaran- tees (e.g., due to system load or because the appropriate medical device has not been connected), then the app manager will not let the clinician start the app in question. If the resources are available, the application manager will reserve those resources so as to

guarantee the required performance to the app. The application manager further detects and flags potential medically meaningful app interactions, since individual apps are isolated and may not be aware which other apps are associated with a given patient.

* *Application database:* Stores the applications installed in the MDCF. Each application contains executable code and requirement meta- data used by the application manager to allocate the appropriate resources for app execution.
* *Clinician service:* Provides an interface for the clinician console GUI to check the status of the system, start apps, and display app GUI elements. Since this interface is exposed as a service, the clini- cian console can be run locally (on the same machine) that is run- ning the supervisor, or it can be run remotely (e.g., at a nurse’s station).
* *Administrator service:* Provides an interface for the administrator’s console. System administrators can use the administrator’s con- sole to install new applications, remove applications, add devices to the device database, and monitor the performance of the system.

#### [Practitioners’ Implications](#_bookmark0)

One can distinguish the following groups of stakeholders in MCPS:

* MCPS developers, including manufacturers of medical devices and integrators of medical information technologies
  + MCPS administrators—typically clinical engineers in hospitals, who are tasked with deploying and maintaining MCPS
  + MCPS users—clinicians who perform treatment using MCPS
  + MCPS subjects—patients
  + MCPS regulators, who certify the safety of MCPS or approve their use for clinical purposes

In the United States, the FDA is the regulatory agency charged with assessing the safety and effectiveness of medical devices and approv- ing them for specific uses.

##### [MCPS Developer Perspective](#_bookmark0)

Dependence of MCPS on software, as well as complexity of software used in medical devices, has been steadily increasing over the past three decades. In recent years, the medical device industry has been plagued with software-related recalls, with 19% of all recalls of medical devices in the United States being related to software problems [Simone13].

Many other safety-regulated industries, such as avionics and nuclear power, operate on relatively long design cycles. By contrast, medical device companies are under intense market pressure to quickly introduce additional features into their products. At the same time, medical devices are often developed by relatively small compa- nies that lack the resources for extensive validation and verification of each new feature they introduce. Model-based development tech- niques, such as the ones described in Section 1.3.3, hold the promise of more efficient verification and validation, leading to shorter devel- opment cycles.

At the same time, many medical device companies complain about the heavy regulatory burden imposed by the FDA and similar regula- tory agencies in other countries. Formal models and verification results, introduced by the model-based development approaches, provide evi- dence that MCPS is safe. Combined with the assurance cases that organize this evidence into a safety argument, these rigorous develop- ment methods may help reduce the regulatory burden for MCPS developers.

##### [MCPS Administrator Perspective](#_bookmark1)

Clinical engineers in hospitals are charged with maintaining the wide variety of medical devices that constitute the MCPS used in patient treatment. Most clinical scenarios today involve multiple medical devices. A clinical engineer needs to ensure that the devices used in treating a patient can all work together. If an incompatibility is discov- ered after treatment commences, the patient may be harmed. Interoperability techniques, described in Section 1.3.4, may help to ensure that more devices are compatible with one another, making the job of maintaining the inventory and the assembly of clinical scenarios easier. This, in turn, reduces treatment errors and improves patient out- comes and, at the same time, saves the hospital money.

##### [MCPS User Perspective](#_bookmark1)

Clinicians use MCPS as part of delivering patient treatments. A specific treatment can, in most cases, be performed with different MCPS imple- mentations using similar devices from different vendors. A primary con- cern, then, is ensuring that clinicians are equally familiar with the different implementations. The concepts of clinical scenarios and virtual medical devices, introduced in Section 1.3.4, can help establish a com- mon user interface for the MCPS, regardless of the specific devices used to implement it. Such an interface would help to reduce clinical errors when using these devices. Furthermore, the user interface can be verified as part of the analysis of the MCPS model, as suggested by [Masci13].

MCPS development must take existing standards of care into con- sideration. Clinical personnel need to be involved in the analysis of the scenario models to ensure that they are consistent with extant clinical guidelines for the respective treatment and are intuitive for caregivers to use.

A particular challenge in modern health care is the high workload faced by caregivers. Each healthcare provider is likely to be caring for multiple patients and must keep track of multiple sources of informa- tion about each patient. On-demand MCPS have the potential to con- trol cognitive overload in caregivers by offering virtual devices that deliver intelligent presentation of clinical information or smart alarm functionality. Smart alarms, which can correlate or prioritize alarms from individual devices, can be of great help to caregivers, by giving a more accurate picture of the patient’s condition and reducing the rate of false alarms [Imhoff09].

##### [Patient Perspective](#_bookmark1)

Of all the various stakeholder groups, patients stand to gain the most from the introduction of MCPS. In addition to the expected improve- ments in the safety of treatments achieved through higher reliability of individual devices and their bedside assemblies, patients would get the benefit of improvements in treatments themselves. These improve- ments may come from several sources.

On the one hand, MCPS can offer continuous monitoring that care- givers, who normally must attend to multiple patients as part of their workload, cannot provide by themselves. Clinical guidelines often require caregivers to obtain patient data at fixed intervals—for example, every 15 minutes. An MCPS may collect patient data as frequently as allowed by each sensor and alert caregivers to changes in the patient’s condition ear- lier, thereby enabling them to intervene before the change leads to a seri- ous problem. Furthermore, continuous monitoring, combined with support for predictive decision making, similar to the system discussed in Section 1.3.5, will allow treatment to be proactive rather than reactive.

Probably the biggest improvement in the quality of care for patients will come with the transition from general guidelines meant to apply to all patients within a certain population to personalized approaches, in which treatment is customized to the individual needs of the patient and takes into account his or her specific characteristics. Personalized treatments, however, cannot be effected without detailed patient mod- els. Such models can be stored in patient records and interpreted by the MCPS during treatment.

##### [MCPS Regulatory Perspective](#_bookmark1)

Regulators of the medical devices industry are tasked with assessing the safety and effectiveness of MCPS. The two main concerns that these regulators face are improving the quality of the assessment and making the best use of the limited resources that agencies have available for performing the assessment. These two concerns are not independent, because more efficient ways of performing assessments would allow regulators more time to conduct deeper evaluations. The safety case technologies discussed in Section 1.3.7 may help address both. The move toward evidence-based assessment may allow regulators to per- form more accurate and reliable assessments. At the same time, organ- izing evidence into a coherent argument will help them perform these assessments more efficiently.

Unit -1[Chapter 2](#_bookmark1)

## [Energy Cyber- Physical Systems](#_bookmark1)

#### [Introduction and Motivation](#_bookmark1)

The electric power industry has the potential to become one of the largest users of cyber-technologies.

Several key hidden problems currently need fixing:

* The increased frequency and duration of service interruption (loss measured in billions)
* Major hidden inefficiencies in today’s system (an estimated 25% economic inefficiency according to the Federal Energy Regulatory Commission [FERC] and supported by case studies)
* Unsustainable operation and planning of high-penetration renew- able resources if the system is operated and planned as in the past
* Lack of seamless electrification of the transportation systems

This, then, raises questions about the major pitfalls in today’s electric power industry in which the use of cyber-technologies is implemented in a piecemeal fashion and never designed with system-level objectives.

#### [System Description and Operational Scenarios](#_bookmark1)

* At present, there are no agreed-on performance metrics necessary to integrate and operate various industry stakeholders at their maximum value.
* Defining the rights, rules, and responsibilities between traditional utilities, on the one hand, and the electricity users, generation providers, and delivery entities, on the other hand, is a work in pro- gress, at best.
* This situation creates major impediments to deploying new technologies so as to realize their full value.
* Another way of thinking about this problem is to note that there are no seamless information technology (IT)–enabled protocols for interfacing the objectives of generation providers, users, and T&D operators within a utility (the company that sells electricity to the final users)—nor, for that matter, between electrically interconnected utilities across larger geographical regions.
* This lack of coordination has major implications for how large-scale wind-and solar-power systems are built and utilized, as well as for the infrastructure necessary to deploy and utilize these new resources.
* Some efforts are under way related to extra-high-voltage (EHV) transmission planning for supporting long-term, large-scale integration of these technologies.
* The operations of the electric power grids that rely on IT have evolved over time.
* They are a result of a careful mix of engineering insights about the power grids and specific-purpose computer algorithms used in an advisory manner by the human operators.
* At present, most modern utility control centers routinely use computer applications, developed using model-based feed-forward techniques, to evaluate and schedule the cheapest predictable power generation for a system where both the demand and the supply vary over time.
* The only utility-level closed-loop feedback coordinated scheme is the automatic generation control (AGC), which is dedicated to balancing quasi-static, hard-to-predict deviations from real power forecasts by adjusting the setpoints of governors on fast-responding power plants.
* The control logic of these governors and other primary controllers (e.g., auto- matic voltage regulators [AVRs]) and some T&D-controllable equipment (e.g., transformers and capacitor banks) is used to ensure local stabilization of frequency and voltage to their setpoints.
* This combination of utility-level power scheduling for predictable power imbalance, quasi-static AGC for frequency regulation, and local primary control of generators and some T&D equipment forms the basis of today’s hierarchical control of power grids.
* More saliently, power grids are complex, large-scale, dynamic networks. Their effective operations and planning require one to view the power grid as a complex system in which energy and information processing are intricately intertwined.
* No ready-to-use tools are available for general CPS that can ensure their large-scale dynamic optimization under uncertainties, nor are there means of ensuring that the dynamics will meet, in a guaranteed way, the performance specifications required for reliability.
* Many general CPS concepts are applicable, but creating formal definitions of utility problems for purposes of applying more general CPS concepts has proved difficult.
* Perhaps the biggest challenge is ensuring provable performance in systems of such high complexity as are seen in electric power grids; this poses major challenges to the overall state of the art in systems engineering. Notably, the system dynamics in the electric power industry are highly nonlinear. The time–space scales over which acceptable per- formance is required are vast and range from milliseconds to decades, and from an appliance or building to the eastern or western U.S. power grid interconnections, respectively.
* Fast storage remains scarce and expensive, and it is still not readily available—a factor that limits the controllability of the grid. In addition, the current systems are not dynamically observable.
* Large-scale optimization under the uncertain- ties inherent in these grids remains a major challenge, though there are targeted efforts under way to introduce more computationally effective algorithms using powerful computers.

#### [Key Design Drivers and Quality Attributes](#_bookmark1)

In the electric power industry, the new trends involve a mix of innovations in physical energy systems (power grid, power electronics, energy resources), communication systems (hardware and protocols), control systems (algorithms and massive computation), and economic and policy systems (regulations and

stimuli).

It has been estimated that just the online IT implementations of these innovations would enable the following benefits:

* + 1. Approximately 20% increase in economic efficiency
    2. Cost-effective integration of renewable energy sources and reduc- tion of emissions
    3. Differentiated quality of service (QoS) without sacrificing of basic reliability of service
    4. Seamless prevention of blackouts
    5. Infrastructure expansion (e.g., generation, T&D, demand side) for maximum benefit and minimum intrusion
* Achieving the system-level objectives by implementing online IT in complex power grids represents a huge intellectual challenge.
* Efforts to define and model the prob- lem of future electric energy systems as a heterogeneous technical, social, and institutional complex system are long overdue.
* To fill this gap, it is necessary to represent in sufficient detail the multilayered and multidirectional dynamic interactions involving these systems, which are driven by often-unaligned societal needs and the needs of various stakeholder groups, and which are constrained by the ability of the energy systems to integrate them according to choice and at value.
* It is essential to design intelligence for T&D operations that would be capable of aligning these goals and, consequently, getting the most out of the available resources, while simultaneously providing acceptable QoS, including resilient response to major uncertainties.
* The CPS technologies will play a key role in the transitioning of the existing system to the new system as new sensors and actuators are deployed into legacy power grids.
* These myriad challenges will require systems thinking if cyber-technologies are to be ultimately designed, implemented, and proven useful in major ways.

##### [Key Systems Principles](#_bookmark1)

* Today’s primarily grid-centric design is rapidly becoming outdated and inadequate.
* It is no longer possible to assume that utilities can forecast and manage demand to the level of coarseness necessary to respond effectively to ever-changing conditions and customers’ preferences, and to account for the effects of intermittent resources as they shape utility-level supply–demand imbalances.
* Similarly, it is no longer possible to assume full knowledge and control of a rapidly growing number of small, diverse distributed energy resources (DERs) connected to the lower distribution grid voltage levels.
* Instead, given different technology developers and owners, it will be necessary to introduce both planning and operating protocols for exchanging information about the functionalities of the DERs equipped with cyber “smarts.”
* In this environment, participants in the electric power industry must be able to differentiate the effects of ordinary buildings and smart buildings, fast-charging and smart-charging electric vehicles (EVs), smart and passive wires, and much more.

###### Sustainable Socio-ecological Energy Systems

* We can consider socio-ecological energy systems (SEES) to be similar to any other socio-ecological systems, comprising resources, users, and governance.
* A smart grid has the key overall function of matching the characteristics of the SEES resources, users, and govern- ance system, which are defined in terms of their temporal, spatial, and contextual properties.
* The more closely these characteristics are aligned, the more sustainable the system will be.

**Table :** *Architectures for Socio-ecological Energy Systems*

|  |  |  |
| --- | --- | --- |
| **Architecture** | **Description** | **Operating Environment** |
| 1 | Bulk SEES | Regulated |
| 2 | Bulk SEES | Restructured |
| 3 | Hybrid SEES | Restructured |
| 4 | Fully distributed SEES | Developed countries |
| 5 | Fully distributed SEES | Developing countries |

* + Notably, the resources, users, and governance system can align these properties either internally in a distributed way or by managing interactions between them via a human-made physical power grid and its cyber-smarts.
  + Posing the problem of CPS design for a sustainable SEES this way sets the stage for introducing meaningful performance specifications for its design.
  + In the Table, Architectures 1 and 2 represent large-scale systems whose major energy resources generate constant power at peak capacity, such as nuclear plants.
  + The governance in place considers these energy services to be a public good and social right; the design of such systems evolves around unconditional service to relatively passive energy users.
  + In the Figure depicting Architecture 1, the larger circles at the top of this figure denote conventional bulk power resources, the smaller circles denote DERs (renewable generation, aggregate responsive loads), and the interconnecting lines represent the transmission grid.
  + Architectures 1 and 2 differ mainly in terms of their governance systems.
  + In Architecture 2, energy resources are generally privately owned and controlled; energy is provided through electricity market arrangements.

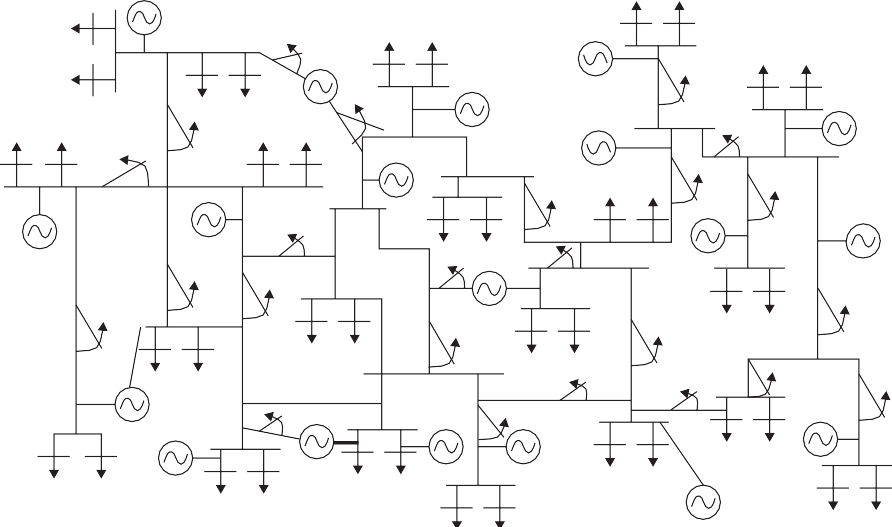


**Figure:** *SEES Architectures 1 and 2. Circles with wiggles on top represent generators, arrows represent end users, and interconnections represent controllable wires.*



**Figure :** *SEES Architecture 3. Circles represent already controlled power plants, arrows represent end users, and the interconnections are controllable wires.*

* Architecture 3, illustrated in another Figure, represents a combination of hybrid resources (large-scale, fully controllable generation and a certain percentage of intermittent resources), a mix of passive and responsive demand, governance requiring high-quality energy service, and compliance with high emission rules.
* Architectures 4 and 5, shown in next Figure, comprise a stand-alone distribution-level micro-grid—for example, an island; a large com- puter data center; military, navy, or air base; or a shopping mall com- pletely disconnected from the local utility.



**Figure 2.3:** *SEES Architectures 4 and 5*

This kind of SEES is supplied by many small intermittent, highly variable DERs, such as rooftop photovoltaics (PVs), bioresources, small-scale wind power generation, small pumped hydropower sources, and the like.

Architectures 4 and 5 have qualitatively different performance objectives for reliability and efficiency.

Architecture 4 is typically characterized by higher requirements for QoS and lower emphasis on cost minimization than Architectures 1 through 3.

In contrast, Architecture 5 is typical of the low-cost, low-QoS systems found in developing countries, in which the objective is to provide electricity services within the financial constraints of electricity users.

Table below, highlights the major differences between the objectives of cyber-physical systems in regulated bulk electric energy systems on one side, and the evolving hybrid and distributed energy systems on the other.

In Architectures 3 through 5, the performance objectives of the system as a whole reflect the system users’ needs and preferences. In contrast, in Architecture 1, and to a lesser extent in Architecture 2, the performance objectives for the power grid are defined in a top- down way by the utility and the governance system.

**Table 2.2:** *Qualitatively Different Performance Objectives in SEES Architectures*

|  |  |
| --- | --- |
| **Performance Objectives in Architectures 1 and 2** | **Performance Objectives in Architectures 3–5** |
| Deliver supply to meet given demand | Both supply and demand have preferences (bids) |
| Deliver power assuming to predefined tariff | QoS given willingness to pay for it |
| Deliver power subject to predefined emission constraints | QoS given willingness to pay for emissions (preferences) |
| Deliver supply to meet given demand subject to given grid constraints | QoS including delivery at value |
| Use storage to balance fast-varying supply and demand | Build storage given grid users’ preferences for stable service |
| Build new grid components for forecast demand | Build new grid components according to longer-term ex-ante contract for service |

###### Critical System-Level Characteristics

* SEES is more sustainable when the characteristics of the users, resources, and governance system are more closely aligned.
* The sustainability of a bulk power system (i.e., Architecture 1) critically depends on system-level optimization of the supply–demand.
* In Architectures 3 through 5, CPS becomes necessary to enable users and resources to manage themselves in a more distributed way and to charac- terize their participation in system-level functions.
* In Architecture 2, social objectives are defined by the governance system (regulators) and implemented by the system operators, planners, and electricity markets based on the specifications identified in system users’ bids.
* CPS design evolving systems of Architectures 4 and 5 needs to be developed so that the temporal, spatial, and contextual characteris- tics of both newly added components and the existing system work well together.
* Similarly, users have their own sub-objectives, which comprise a combination of cost, environmental, and QoS preferences.
* It is important to note that the type of architecture largely determines the contextual preferences since the governance characteristics are qualitatively different.

Coal G Bus 1

Price- responsive loads

Natural Gas G

PHEV

G Wind Bus 2

Bus 3

Photovoltaic G

PHEV

**Figure 2.4:** *A small-system illustration of the evolving SEES*



##### [Architecture 1 Performance Objectives](#_bookmark1)

* The main performance objective of Architecture 1 is network-centric. Each utility is required to supply energy to its users all the time and at the lowest cost possible.
* This is achieved by scheduling the available generation *PG*  and delivering this power to supply the system-level utility load.
* The transmission investment decisions are often based on analyses for the worst-case scenario and are not optimized.
* The system demand *PD* is assumed to be known, and planning of *“Generator Capacity’* and ‘Transmission Capacity’is done to have sufficient capacity during the worst-case equipment failures in which one or two large generators and/or transmission components are disconnected from the grid. This case is known as the (*N* – 1) or (*N* – 2) reliability criterion.
* Notably, so-called performance-based regulation (PBR) represents an effort to provide incentives to utilities to consider innovation that might be beneficial in reducing the overall cost of service, while meeting the customers’ QoS expectations.
* A PBR design for complex energy networks remains an open problem at present.

###### Systems Issues in Architecture 1

They can be classified as follows:

* *Nonlinear dynamics-related problems:* As yet, there are no nonlinear models that would support the sensor and actuator logic needed to ensure the desired performance both during normal operations and during abnormal conditions.
* *Use of models that do not control instability adaptively:* All controllers are constant gain and decentralized (local). The system is not fully controllable or observable.
* *Time–space network complexity-related problems:* Problems occur when attempting to avoid fast dynamic instabilities following major equipment outages by means of conservative resource scheduling. The system is generally operated inefficiently by requiring stand- by fast reserves just in case large equipment fails.
* New best-effort solutions are generally not acceptable by the industry and are viewed as threats to reliable operations.

###### Enhanced Cyber Capabilities in Architecture 1 Systems

* + Advanced metering infrastructures (AMIs) are used for information gathering which is used to implement adaptive load management (ALM) that benefits both customers and utilities.
  + As system demand becomes less predictable, the value of CPS in enabling just-in-time (JIT) and just-in-place (JIP) supply–demand balancing will only increase.
  + A robust state estimator (SE) and scheduling software for adjusting other controllable equipment as the loading conditions vary, could avoid—or at least lower—the high capital costs of the large-scale equipment that is currently viewed as necessary for providing reliable service.

###### CPS Design Challenges for Architecture 1

* Innovations and the adoption of new cyber-based solutions have been rather slow.
* One of the major computational challenges comes from the highly intertwined temporal and spatial complexities within the large- scale electric power grid.
* The controllable equipment has its own dynamics, and the power grid is spatially complex and nonlinear.

###### 

###### CPS Design Challenges for Architecture 2

* In this type of system, however, the generation costs result from non-utility-owned bids and may not be the same as the actual generation costs.
* Also, at least in principle, load-serving utilities are being formed as competition to today’s utilities and are offering to provide service to groups of smaller users by participating in the wholesale market.
* In this scenario, the objective of these load-serving entities (LSEs) is profit maximization by managing load served.
* The main challenge in Architecture 2 is to develop a cyber design that ensures electricity markets are consistent with power grid operations.

###### CPS Design Challenges for Architectures 3–5

* The system-level objectives of the emerging Architectures 3 through 5 are not well defined at present.
* Architectures 4 and 5 are basically at the green-field design stage;
* Large electricity users separating from the utility grid and becoming effectively stand-alone micro-grids; this trend is taking place in smart data centers, military bases, and navy and airspace centers.
* Some groups comprising relatively small users managed by energy service providers (ESPs) are beginning to consider the benefits of stand-alone resources and lower dependence on the local utility providers.
* They can be completely disconnected from the backbone power grids, or they can be connected to them for ensuring reliable service--- referred to as micro-grids.
* Very little formal mathematical modeling of the emerging architectures, as complex large-scale dynamical systems, has been performed to date.
* Instead, much effort has been devoted to modeling stand-alone (groups of) components, such as wind power plants, smart buildings, solar power plants, and their local controllers and sensors.

##### [A Possible Way Forward](#_bookmark1)

* + No single method will be sufficient to support truly resilient and efficient operations of future power grids.
  + More generally, proactive participation of electricity users in power grid supply–demand balancing for environmental and economic reasons is likely to make it almost impossible for utilities to rely on the same system demand forecasting techniques and schedule generation as they used in the past.
  + Many of the DERs and large loads that are self-sufficient during normal operations might still wish to call upon utility support when their own resources are not available/on break down.
  + This pattern puts major pressure on utilities as the last-resort providers of electricity service.
  + To induce efficiency and robustness, cyber-enabled aggregation of resources will be essential.

#### [Cyber Paradigm for Sustainable SEES](#_bookmark1)

A DyMonDS(Dynamic Monitoring and Decision System) orm as a basis for new cyber design and implementation to overcome the previously described problems in the electric power industry.

Hydro Gen. and Pumped Storage

DYM

Generators

DYM DYM

Large Scale Wind Farm

DYM



Central Mesh Network Distribution Network

Future network additions

SCADA Data Flow

DYM Proposed DyMonDS

Data Flow for present grid users

DYM Distributed

Generation

DYM Industrial Loads

DYM Storage and

Storage/generation



Substation

DYM

Industrial

Transmission lines

DYM

Central Mesh Network

Control center

DYM

Medium Size D.G.

Battery Storage

Substation

Transmission lines

DYM

D

YM

Industrial

Load

DYM

Substation

Substation

DYM

Load

Residential Load

Residential Load

Residential Load

Residential Load

DYM

DYM

DYM

DYM

DYM

DYM

Commercial Load

DYM

DYM

DYM

PHEV Fleet

DYM

DYM

1. Grandma’s House of the
2. House in cold
3. Green factory

future

1. House in warm location
2. House in warm location with extreme grid conditions

location

with wind farm

**Figure 2.5:** *CPS electric energy system with its embedded DyMonDS [Ilic11, Ilic11a]*

* Two CPS processes can be observed: (1) the physical (groups of) components with their own local DyMonDS, and (2) the information and energy flow between the (groups of) components, shown in dotted multi-directional dotted lines.
* DyMonDS effectively represents (Groups of) physical components with their embedded sensing, learning of system signals, and local decision making and control.
* Components respond to the pre-agreed binding information signals and send pre-agreed information back to the system according to well-defined protocols.
* A smart and strong grid is strongly dependent on the type of SEES architecture it is supposed to serve.
* Figure below is a sketch of the proposed DyMonDS-based end-to-end CPS grid enabling sustainability of a socio-ecological energy system (SEES).
* Align the temporal, spatial, and contextual characteristics with the objectives of the resources, users, and governance system. Then we can design the CPS grid with well-defined objectives, and the value of many diverse grid technologies can be quantified.

Physical network connecting energy generation and consumers

Needed to implement interactions

Human-Made Grid

Resource system (RS)

Generation (RUs)

Electric energy users (U)

Energy SES

Sensors Communications Operations

Decisions and control

Protection

Human-Made ICT

*Smart grid: end-to-end CPS for enabling sustainable SEES*

##### 

##### [Physics-Based Composition of CPS for an SEES](#_bookmark1)

* A new physics-based approach to cyber-network composition in support of physical energy systems is proposed.
* The dynamics of each physical module is characterized by the module’s own state variables *xi*(*t*), with the number of states being *ni*; the local primary controllers injecting input signals *ui*(*t*); external disturbances *Mi*(*t*); and interaction variables *zi*(*t*).
* The composition challenge is the one of combining all modules into a complex interconnected system;
* The open-loop dynamics of an interconnected SEES are determined by the external inputs *Mi*(*t*) to all modules, and by the initial state of all modules *xi*(0).
* Closed-loop dynamics are driven by changes in set- points of local controllers *u*ref(*t*) and the local control *u*(*t*) reacting to a combination of deviations in local outputs *yi*(*t*) = *Cixi*(*t*) and interaction variables *zi*(*t*) from the desired values.
* System regulation and local control designs, Together with the System Control and Data Acquisition System (SCADA), they form the cyber network for an SEES.
* To formalize the systematic composition of a cyber network, we first observe that the actual continuous dynamics of states *xi*(*t*) has physics- based structure.
* In particular, the dynamics of any given module depends explicitly on the dynamics of the neighboring states only, and not on the states of farther away modules.
* The transformed state space for each module comprises the remaining (*ni* – 2) states and the interaction variable whose physical interpretation is the net energy stored in the module and the rate of stored energy exchanged between the module and the rest of the system.

*Pointer*

**Figure 2.8:** *General representation of a CPS module within a SEES*

Module I

(Subclass)

Module II (Subclass)

*Pointer*

Large Disturbance/ Emergency happens

Or

Equipment fails to work

Learned Structure Data

Communicated Structure Data

Equipment Status Flag

Exogenous Input

Initial States

Parameters

Type

Real Time Data

**C-Database III**

**L-Database III**

**Algorithms**

**C-Database II**

**L-Database II**

**Functions**

**C-Database I**

**L-Database I**

**MODEL**

* The transformed state space for each module comprises the remaining (*ni* – 2) states and the interaction variable whose physical interpretation is the net energy stored in the module and the rate of stored energy exchanged between the module and the rest of the system.
* This, in turn, defines the relevant cyber variables, which are the minimum necessary for characterizing the dynamics of the complex grid.
* This multilayered representation is shown in Figure 2.9. In this model, the learned variables in Figure 2.8 for each zoomed-in module are the interaction variables of the neighboring modules, and the communicated variables between the module and the rest of the system are the module’s own interaction variables. This modeling step encourages one to view electric energy systems as genuine cyber-physical systems; their physics defines the cyber-structure (the variables that must be known and communicated), and the cyber performance is designed to support the physical performance. The open-loop dynamics can be

85

2.4 Cyber Paradigm for Sustainable SEES

*Pointer*

*Pointer*

*Pointer*

Real Time Data

Real Time Data

x

Real Time Data Real Time Data



Equipments Status Flag

Exogenous Input

Initial States

Parameters

Type

Equipments Status Flag

Exogenous Input

Initial States

Parameters

Type

Equipments Status Flag

Exogenous Input

Initial States

Parameters

Type

Equipments Status Flag

Exogenous Input

Initial States

Parameters

Type

**C-Database II**

**L-Database II**

**Functions**

**C-Database I**

**L-Database I**

**MODEL**

**Algorithms**

**C-Database II**

**L-Database II**

**Functions**

**C-Database I**

**L-Database I**

**MODEL**

Simplified representation using interaction variables

**C-Database III**

**L-Database III**

**Algorithms**

**C-Database II**

**L-Database II**

**Functions**

**C-Database I**

**L-Database I**

**MODEL**

**C-Database III**

**L-Database III**

**Algorithms**

**C-Database II**

**L-Database II**

**Functions**

**C-Database I**

**L-Database I**

**MODEL**

Module I (Subclass)

Module II (Subclass)

Large Disturbance/ Emergency happen

Or

Equipment fails to work

Module I (Subclass)

Module II (Subclass)

*Pointer*

*Pointer*

Large Disturbance/ Emergency happen

Or

Equipment fails to work

Learned Structure Data

Communicated Structure Data

Learned Structure Data

Communicated Structure Data

y

y

x

Module I (Subclass)

*Pointer*

Large Disturbance/ Emergency happen

Or

Equipment fails to work

Module I (Subclass)

Module II (Subclass)

Module II (Subclass)

*Pointer*

Large Disturbance/ Emergency happen

Or

Equipment fails to work

Learned Structure Data

Communicated Structure Data

Learned Structure Data

Communicated Structure Data

**L-Database III**

**C-Database III**

**L-Database III**

**Algorithms**

**C-Database III**



Module I (Subclass)

Module II (Subclass)

*Pointer*

Large Disturbance/ Emergency happens

Or

Equipment fails to work

Learned Structure Data

Communicated Structure Data

Real Time Data

Detailed internal representation using the full set of internal states

c

a

r

r

q

d

b

Equipment Status Flag

Exogenous Input

Initial States

Parameters

Type

**C-Database III**

**L-Database III**

**C-Database II**

**L-Database II**

**C-Database I**

**L-Database I**

*Pointer*

*Pointer*

y

x

y

x

**Figure 2.9:** *Interaction variables–based CPS composition within an SEES*

* composed by exchanging information about the interaction variables. This interaction supports the efficient numerical methods needed to simulate these otherwise very complex grids
* Given sufficient control of each module to keep the internal states stable, the most straightforward distributed control responds to a combi- nation of dynamics of the internal variables and of the interaction varia- bles.
* This amounts to fully decentralized competitive control when using the transformed state space.
* The cyber aspects will amount to fast sampling in support of often high-gain local control and no communications; full system synchronization is achievable this way.
* This cyber- structure would be very effective in an ideal world, such as in Architecture 4, in which all dynamic components, including wires, have fast typically power-electronically switched controllers.

##### [DyMonDS-Based Standards for CPS of an SEES](#_bookmark1)

* New **Dynamic Monitoring and Decision Systems** frame work can be best understood by using the new modeling approach in which modules are characterized by their internal and interaction variables.
* In DyMonDS embedded in physical modules, the design enables modules to be inserted into an existing grid in a plug-and-play manner as long as they meet the standards
* The multilayered modeling and control design approach sets the basis for the standards that must be met by different modules.
* The simplest approach is to require each module to meet the dynamic standards such that the interactions with the rest of the system are cancelled in closed-loop dynamics.
* This approach is probably acceptable in Architectures 4 and 5, which comprise many small distributed components, each of which has a local cyber-structure.
* For example, a PV panel might have local storage and fast power electronics capable of cancelling the effects of the stored energy exchange (interaction variables) coming from the rest of the system to the PV.
* Such a PV would have sufficient control and storage that it could behave as an ideal AC voltage source, at least as seen by the rest of the system;
* In contrast, Architectures 1 through 3 require more complicated standards for plug-and-play dynamics. Notably, not every dynamic component in these architectures has a controller.
* This standard requires cooperative and coordinated control within each group of modules, and minimal or no coordination among the groups.
* The interactions between the two control areas are measured in terms of inadvertent energy exchange (IEE), which is determined by the area control error (ACE) power imbalance; this is the underpinning of today’s AGC.
* A dynamic interaction variable associated with a group of components responsible for ensuring there are no instability problems is a necessary.
* Detailed derivations have shown that the dynamics of these interaction variables is determined by the interaction variables themselves and by some states of the neighboring modules.
* Therefore, it becomes possible to sense and control the interaction variables in a distributed way as well, because their dynamics are fully determined by the locally measurable variables.

###### The Role of Data-Driven Dynamic Aggregation

* If the effect of interaction variables dominates the effect of local modular dynamics, it becomes necessary to design complex high-gain controllers locally to cancel the interactions and make the modules behave as if their closed-loop dynamics have become weakly coupled.
* This approach generally calls for lots of local sensing and fast controllable storage, and effectively no communications.
* The overwhelming complexity is managed by closing the loop and making the modules and layers self-sufficient.
* They can still achieve equally good performance as if the coordination of interaction variables is done for modules with prefixed boundaries, as is common in today’s hierarchical systems.
* The physics at play ultimately determine the near-optimal balance between the complexity of the local cyber-structure and the communication requirements for coordinating interaction variables.

###### Coordination in Systems with Predefined Subsystems

* Today, it is very difficult to dynamically aggregate groups of users in a bottom-up fashion so that their closed-loop dynamics is very loosely coupled with the other groups of users.
* When an electric power system loses a large generator or a transmission line, the relevant model is inherently nonlinear.
* One possible solution is to use adaptive high-gain control of the remaining components, such that the system appears linear and is differentially flat in the closed loop.
* The major challenges are how to dynamically aggregate many components into a single intelligent balancing authority (iBA) for minimizing the cost of storage.
* Determining when conventional fast controllers shaping very little storage in reactive components of Flexible AC Transmission Systems (FACTS) would have to be supplemented by the real power energy storage devices, such as various batteries and flywheels, is a challenging control design problem at present.
* Implementation of such an approach makes it possible to define standards for dynamics of such iBAs, and the system can be operated in a plug-and- play manner by the weakly coupled iBAs.
* The Figure (a) depicts the IEEE 24-node RTS system and the dynamic response of power flows in the area close to the loss of generator at node 1 with iBA-based high-gain control (constant gain AVRs and PSSs).
* Figure (b) shows the system response without carefully aggregated iBAs; the dynamic response to the same fault with the closed-loop nonlinear controller is also shown.
* While the system’s response to losing generator oscillates with the existing excitation controllers, the high-gain nonlinear feedback-linearizing excitation controllers would stabilize the closed dynamics of the power plants in the iBA, almost completely eliminating the inter-area oscillations with the rest of the system.
* It known that the choice of energy function for keeping the system synchronized is critical under these conditions. Here again, the selection of the energy function is fundamentally based

on the physics:



18

21

22

23

16

19

20

15

14

24

11

13

intelligent

Balancing 12 Authority

(iBA)

3

9

10

6

4

5

8

1

2

7

6

Pe1 Pe2 Pe7

Pe13

Pe23

5

4

3

2

1

0

1

2

30

1 2 3 4 5 6 7 8 9 10

**Figure 2.10:** *Dynamic response with high-gain iBA distributed control*

17



18

21

22

23

16

19

20

15

14

13

24

11

12

3

9

10

4

5

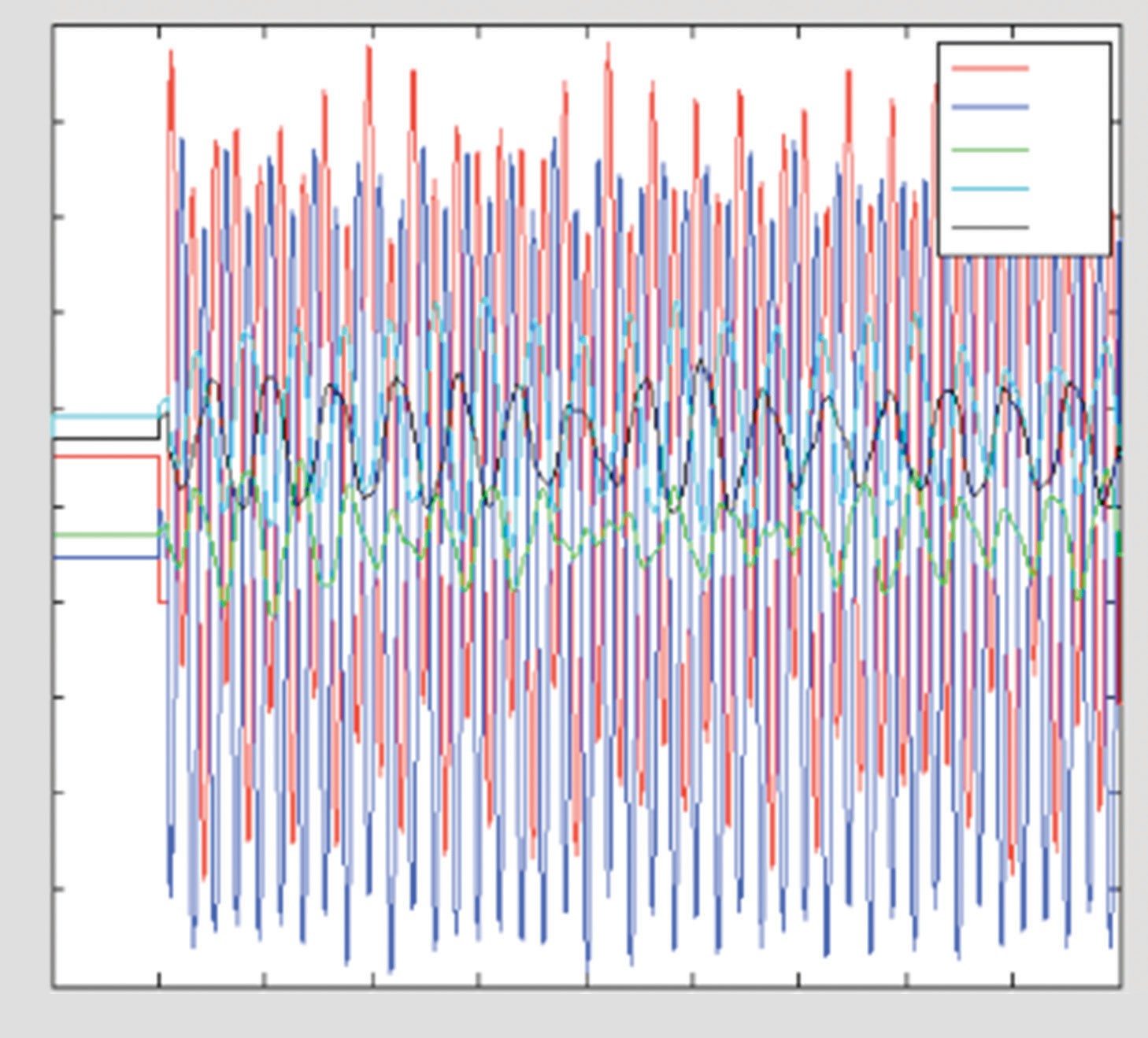
1

2

7

6

8



6

5

4

Pe1 Pe2 Pe7

Pe13 Pe23

3

2

1

0

1

2

3

4

0 1 2 3 4 5 6 7 8 9 10

**Figure 2.11:** *Inter-area oscillations in the system with conventional controllers*

* When system disturbances occur on a continuous basis, the well-established hierarchical electric power system modeling of today cannot cope effectivel
* The upper layer in the cyber-structure then senses and communi- cates the dynamics of interaction variables between different system modules; the information exchange required is defined in terms of these interaction variables.
* The local variables and their local controllers are fundamentally distributed as long as the effects of interactions do not overpower the local dynamics of distributed modules. Figure below provides a zoomed-in sketch of a local distributed DyMonDS for given interaction variables, and a zoomed-out sketch of the higher-level coordination of the modules.
* The designs use the structure-preserving modeling framework based on the modularization of the system into two layers—the coordinating layer and the component-level layer.
* Researchers are currently investigating generalizations of this multilayered approach to ensuring transient stability of complex power grids in response to large faults and/or sudden large-amplitude wind disturbances; this approach may become critical for systematic integration of power-electronically switched
* controllers for stabilization of system frequency and voltage
  1. MW



1.5 MW 1

4.2 MW

8.2 MW

1 MW

2

3

Fault

12.4 MW

**Figure 2.12:** *A short-circuit fault in a small power system [Cvetkovic11, Ilic12]*

0.05

0.01

rotor frequencies

ω1 ω2 ω3

0.005

frequency(pu)

0

0.005

0.01

0 1

2 3 4 5

time(s)

0.05

0.01

rotor frequencies

0.005

ω1 ω2 ω3

frequency(pu)

0

0.005

0.01

0 1

2 3 4 5

time(s)

**Figure 2.13:** *Frequency response without and with ectropy-based controller*

Zoom-in Zoom-out

*zK I*



Sk

SI

...

Sj



*z*

...

*zJ*

**Figure 2.14:** *Multilayered approach to coordinating interactions [Ilic12a]*

##### [Interaction Variable–Based Automated M odeling and Control](#_bookmark1)

* Deriving dynamic models that capture the physical phenomena of interest in sufficient detail has long been the major challenge.
* Adopting the perspective that future energy systems will consist of multilayered interactive modular systems allows for representing highly diverse modules and their local DyMonDS in as much detail as desired and defining input/output functionalities in terms of interac- tion variables between the modules for shaping the dynamics of the interconnected system.
* Coordination of interaction variables generally requires much coarser models.
* An automated modular approach for deriving the state space model was implemented. Using this approach, the system is divided into modules:
* The state space model for each module is determined separately, and then the state space models from each module are combined in an automated procedure. This modular approach is particularly useful for power systems because large systems contain many of the same types of components.
* Additionally, this approach can be used to obtain dynamic models suitable for control design.
* In particular, it can be used to derive a model that includes Hamiltonian dynamics.
* The resulting model significantly simplifies controller design by explicitly capturing the dynamics of the outputs of interest, such as the Hamiltonian, which is the total accumulated energy.
* Most recently, this automated modeling approach has been used to design power- electronically controlled switches of flywheels to ensure synchronism in power grids with induction-type wind power plants, which are otherwise vulnerable to prolonged fast power imbalances.

#### [Practitioners’ Implications](#_bookmark1)

* The new SCADA framework allows, operators and planners make decisions based on proactive and binding information exchange with system users;
* This makes it possible to account for choices based on value, while simultaneously meeting difficult societal goals.
* The information exchange is repeated periodically, as decisions are being made for very long future, or for closer to real time.
* This process becomes a win–win situation for all, and future uncertainties are distributed over many stakeholders according to the risks they are willing to bear and according to the expected value from the selected technologies.
* Ultimately, they lead to qualitatively new, reliable, efficient, and clean services, and those technologies that bring value survive beyond the subsidy stages.

##### [IT-Enabled Evolution of Performance Objectives](#_bookmark1)

* Multilayered modeling that requires coordination of interaction varia- bles, as described earlier, assumes that boundaries of cooperative (groups of) components with common objectives are given; in addition, the information exchange patterns within the complex system are con- strained by various governance rules, rights, and responsibilities.
* Over the longer term, one might consider bottom-up interactive dynamic aggregation for optimizing common goals, as well as for providing quantifiable feedback to the governance system regarding the effects of its rules on system performance.
* With this approach, performance objectives are shaped through interactions, and the system evolves into a more sustainable SEES. Much research remains to be done to extend interaction variables–based modeling and design so that they influence the governance system for the electric power industry, the preferences of system users, and the responsiveness of the T&D grid at value.

##### [Distributed Optimization](#_bookmark1)

* Optimizing resources in highly uncertain environments and ensuring that intertemporal and spatial dependencies (constraints) are taken into account are computationally extremely challenging.
* To overcome this complexity, our DyMonDS framework can be used to internalize time-scale and uncertainty- related complexities that might prevent controllable generation and transmission from responding as needed.
* In particular, this model proposes to embed sequential decision making into local DyMonDS that face uncertainties, with the aim of enabling the functionalities of controllable equipment needed for online resource management by the utility’s control center without imposing temporal constraints.
* The simplest example might be a power plant that commits to a time- varying range of power capacity limits that it can produce without requesting the system operator to observe its ramp rates; the ramp rates are accounted for and internalized when the commitment is made.
* The interactions between system users and the system operator can be either pointwise or functional.
* Most of the distributed optimization approaches are fundamentally pointwise in nature:
* Each user performs the necessary optimization for the assumed system conditions, and the optimal request for power quantity is submitted.
* The system operator then collects the requests optimized by the users, and computes the resulting power mismatch.
* The electricity price is updated accordingly. The larger the supply–demand shortage, the higher the system price is in the next iterative step. Conversely, the larger the supply–demand excess, the lower the price is in the next iterative step.
* However, the implementation of pointwise distributed balancing of supply–demand would require a great deal of communication between the system users and the operator.
* As an alternative, the DyMonDS framework is based on exchanging functions between layers (demand and supply functions for generation and even transmission).
* This approach eliminates the need for multiple iterations between the modules and makes it feasible to arrive at the consensus relatively rapidly.