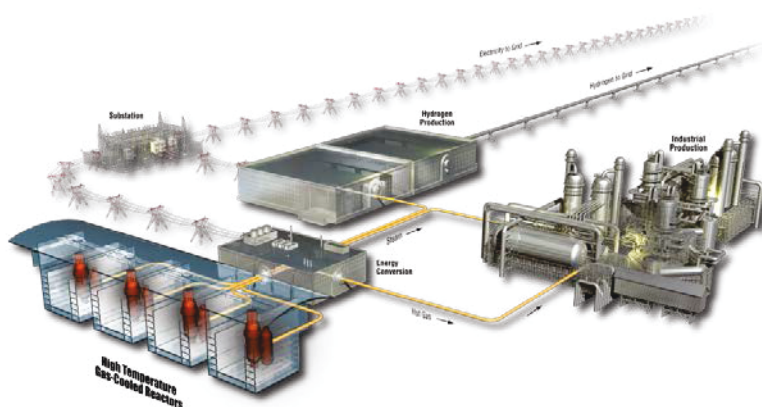


Strategy and Gaps for Modeling, Simulation, and Control of Hybrid Systems

Cristian Rabiti
Humberto E. Garcia
Rob Hovsopian
Robert A. Kinoshita
George L. Mesina
Shannon M. Bragg-Sitton
Richard D. Boardman

April 2015

The INL is a
U.S. Department of Energy
National Laboratory
operated by
Battelle Energy Alliance



DISCLAIMER

This information was prepared as an account of work sponsored by an agency of the U.S. Government. Neither the U.S. Government nor any agency thereof, nor any of their employees, makes any warranty, expressed or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness, of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. References herein to any specific commercial product, process, or service by trade name, trade mark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the U.S. Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the U.S. Government or any agency thereof.

Strategy and Gaps for Modeling, Simulation, and Control of Hybrid Systems

Cristian Rabiti
Humberto E. Garcia
Rob Hovsopian
Robert A. Kinoshita
George L. Mesina
Shannon M. Bragg-Sitton
Richard D. Boardman

April 2015

**Idaho National Laboratory
INL ART Program
Idaho Falls, Idaho 83415**

<http://www.inl.gov>

**Prepared for the
U.S. Department of Energy
Office of Nuclear Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517**

INL ART Program

Strategy and Gaps for Modeling, Simulation, and Control of Hybrid Systems

INL/EXT-15-34877
Revision 0

April 2015

Author:



Cristian Rabiti
Principle Investigator

04/28/2015

Date

Approved by:



Shannon M. Bragg-Sitton
Nuclear Hybrid Energy System Co-Lead

04/29/2015

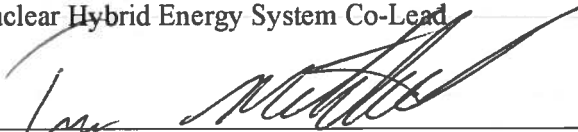
Date



Richard D. Boardman
Nuclear Hybrid Energy System Co-Lead

4/29/15

Date



Travis R. Mitchell
INL ART TDO Relationship Manager

4/30/15

Date



Kirk W. Bailey
INL ART TDO Quality Assurance

4.30.2015

Date

ABSTRACT

This report establishes a strategy for modeling, simulation, and control of candidate hybrid energy systems. Modeling, simulation, and control are necessary to design, evaluate, and optimize the systems' technical and economic performance. This report first establishes modeling requirements to analyze candidate hybrid systems (a strict definition of "hybrid system" will be also provided). Modeling fidelity levels are based on the temporal scale, real and synthetic data availability or needs, solution accuracy, and output parameters needed to evaluate case-specific figures of merit (FOMs). The associated computational and co-simulation resources needed are established, including physical models when needed, code assembly and integrated solutions platforms, mathematical solvers, and data processing.

This report first describes the FOMs, systems requirements, and constraints necessary to characterize the grid and hybrid system behaviors and market interactions. Grid reliability assessment metrics and effective cost of energy (ECE), as opposed to the standard levelized cost of electricity (LCOE), are introduced as technical and economic indices for integrated energy system evaluations. Financial assessment methods are subsequently introduced for evaluating nontraditional, hybrid energy systems. Algorithms for coupled and iterative evaluation of the technical and economic performance are subsequently discussed.

This report further defines modeling objectives, computational tools, solution approaches, and real-time data collection and processing (in some cases using real test units) that will be required to model, control, co-simulate, and optimize: (1) energy system's components (e.g., power generation unit, chemical process, electricity management unit), (2) system domains (e.g., thermal, electrical or chemical energy generation, conversion, and transport), and (3) system control modules. Controlling and co-simulating complex, tightly coupled, dynamic energy systems requires multiple controls and simulation tools, potentially developed in several programming languages and resolved on separate time scales. Whereas further investigation and development of hybrid concepts will provide a more complete understanding of the joint computational and physical modeling and control needs, this report highlights areas where control and co-simulation capabilities are warranted. The current development status, quality assurance, availability, and maintainability of control and simulation tools available for hybrid systems modeling are presented. Existing gaps in the modeling, simulation, and control toolsets and development needs are subsequently discussed. This work will feed into broader efforts to design, develop, and demonstrate hybrid energy systems.

CONTENTS

ABSTRACT	vii
ACRONYMS.....	xii
1. INTRODUCTION.....	1
2. DEFINING THE SIMULATION GOALS	1
2.1 Figures of Merit	3
2.2 The Effective Cost of Energy (ECE) as an Initial Economic Metric.....	4
2.3 Evaluation of the Economical Performance of Each Subsystem.....	5
2.4 Economic Analysis to Assess the Natural Evolution of the System.....	5
2.5 Summary of the Performance Analysis Approach for Hybrid Systems	5
3. DEFINITION OF SIMULATION NEEDS	6
3.1 ECE Analysis	6
3.1.1 Software Requirements.....	6
3.1.1 Necessary Algorithms.....	8
3.1.2 Necessary Grid Component Models.....	11
3.2 Profitability Analysis of Grid Components	12
3.3 Evolution Analysis.....	13
4. SIMULATION FRAMEWORK	13
4.1 Functional Mockup Interfaces and Functional Mockup Unit.....	16
4.1.1 Construction of C-Binding Interfaces.....	16
4.1.1 Platform with Direct Interface to FMU	17
4.2 Ptolemy	23
4.2.1 Building Controls Virtual Test Bed.....	24
4.2.2 Overview.....	24
4.3 RAVEN.....	25
4.3.1 Overview.....	26
4.4 Modelica	26
4.5 RTDS and PSCAD.....	28
4.6 RELAP5-3D.....	30
4.6.1 General Code Simulation Capabilities.....	31
4.6.2 Computational Runtime.....	32
4.6.3 Code Pedigree.....	32
4.7 Aspen	32
5. SOFTWARE DEVELOPMENT PROCESS	33
5.1 Software Lifecycle Management and Quality Assurance.....	34
6. CONCLUSION	36
7. REFERENCES	37

8. APPENDIX	41
Appendix A Summary of Modeling Tools	43

FIGURES

Figure 1. Time scales for the specified stabilizing energy reserves. Several approaches work to assess the different grid reliability requirements. ¹⁴ . Generally, the most-accurate and advanced methodologies use a method of statistical analysis of the time series of the load and renewable supplies (e.g., wind and solar).....	7
Figure 2. Simulation flow.	8
Figure 3. Possible utilization of a reduced order model for optimization.	10
Figure 4. Slave-master code structure in presence of multiple types of language.....	11
Figure 5. Possible configuration where RAVEN is used as a driver for a statistical analysis using a distributed system of RTDS.	15
Figure 6. Translation of a generic model to be used under the FMI/FMU protocol.	15
Figure 7. Graphical representation of cross-linking capabilities of the BABEL library (source: https://computation.llnl.gov/casc/components/index.html#page=home).	16
Figure 8. Graphical representation of the hierarchical structure underneath a composite actor.	23
Figure 9. Limit surface representation in a two-dimensional parametric space.	26
Figure 10: Top-level model for a nuclear hybrid energy system in Modelica.	28
Figure 11. Simulation setup with RTDS and integrated HES model.	29
Figure 12. Simulation setup with RTDS and integrated HES model.	30
Figure 13. High temperature steam electrolysis support flow sheet with electrolysis subflow sheet.	33

TABLES

Table 1. Correspondence between tasks, corresponding algorithm type and current development status.	9
Table 2. Compatibility table for the Functional Mockup Unit (FMU) and Functional Mockup Interfaces (FMI) protocols.....	18

ACRONYMS

ABWR	advanced boiling water reactor
ACE	Area Control Error
AECE	avoided effective cost of electricity
APWR	advanced pressurized water reactor
ART	Advanced Reactor Technologies
ATWS	Anticipated Transient Without Scram
BCVTB	Building Controls Virtual Test Bed
BWR	boiling water reactor
CANDU	Canada Deuterium Uranium
CHES	Center for Hybrid and Embedded Software Systems
CHiL	controller hardware-in-the-loop
CiL	controls-in-the-loop
DA	developmental assessment
DOE	U.S. Department of Energy
ECE	effective cost of energy
EMT	ElectroMagnetic Transient
FMI	Functional Mockup Interfaces
FMU	Functional Mockup Unit
FOM	figure of merit
GiL	grid-in-the-loop
GUI	graphical user interface
HES	hybrid energy systems
HiL	hardware-in-the-loop
INL	Idaho National Laboratory
LACE	levelized avoided cost of electricity
LCOE	levelized cost of electricity
LOCA	loss of coolant accident
LOLP	loss of load probability
MOOSE	Multi-physics Object Oriented Simulation Environment
NGNP	Next Generation Nuclear Plant
NREL	National Renewable Energy Laboratory
PHiL	power hardware-in-the-loop
PSCAD	Power System Computer Aided Design

PVM	parallel virtual machine
PWR	pressurized water reactor
QA	Quality Assurance
QLD	Quality Level Determination
RTDS	Real Time Digital Simulators
SMR	small modular reactor
SNAP	Symbolic Nuclear Analysis Package
VHTR	very high temperature reactor

Modeling, Simulation and Control Gap Analysis Report

1. INTRODUCTION

This report establishes a strategy to model, simulate, and control candidate hybrid energy systems to evaluate and optimize the systems' technical and economic performance, and simultaneously develop functionally realistic systems' control schemes. Modeling and simulation gaps to achieving the desired simulation framework are identified. The simulation of hybrid energy systems first requires definition of quantitative metrics by which these systems can be optimized and evaluated. Section 2 provides a derivation of hard metrics that can be numerically assessed via computational simulations based on overall, qualitative performance metrics partially identified in the July 2014 workshop on nuclear renewable hybrid energy systems.¹ Careful review of these metrics suggests three types of analyses are needed to evaluate hybrid energy system concepts: economic performance, grid reliability, and system resilience. Section 3 identifies the high-level needs for each of these analyses in terms of algorithms, model development, controls, and data. Section 4 describes the simulation environment, some possible software candidates, and possible gaps.

Simulation and control needs are clarified and a set of candidate analysis tools is discussed. The possible use of modeling and simulation languages and environments such as RELAP5-3D,² RELAP-7,³ RAVEN,⁴ Multi-physics Object Oriented Simulation Environment (MOOSE),⁵ MATLAB®/Simulink® and its related toolboxes,^{6,7} Modelica,⁸ Aspen,⁹ Ptolemy,¹⁰ Power System Computer Aided Design (PSCAD) and Real Time Digital Simulators (RTDS®),¹¹ Dymola¹² was considered. Appendix A summarizes the characteristics of each tool considered.

Data collection and pre-processing and post-processing tools are discussed in conjunction with approaches or techniques to couple a given set of the tools to perform multi-physics, multi-domain simulations. The significance of imbedded system control functions and system optimization is also described. It is recognized that, when possible, the use of real-time data feeds from active process units (referred to as hardware-in-the-loop [HiL], grid-in-the-loop [GiL], and controls-in-the-loop [CiL]) should be used for model validation and verification. This will be discussed in a separate validation plan. The overall co-development and co-simulation of models and physical systems will be addressed in a subsequent Roadmap planning document (Technology Development Roadmap milestone; the complete draft Roadmap will be distributed for review in December 2015). This report partially coordinates with a preliminary assessment of two regional hybrid energy system concepts developed for evaluation using a foundational modeling, simulation, control, and optimization framework. Modelica and MATLAB were applied in this previous work to make a preliminary estimate of the economic and technical performance of these cases.

The analysis of each tool and development environment takes into account long-term maintainability, potential costs for necessary development, and synergies with other ongoing programs. Other operational aspects, such as the necessity to coordinate efforts across multiple laboratories and universities, are also considered.

2. DEFINING THE SIMULATION GOALS

The energy market supply is under pressure to achieve competing goals, such as minimizing the cost of electricity and greenhouse gas emissions while achieving grid resilience and reliability and meeting regulatory requirements. Simulating the overall U.S. energy market is outside the scope for the current project; therefore, the simulation, strategies, and associated gap analysis discussed here apply to what could be identified as regional markets.

This document uses the terms grid, system, subsystem, physical, and computational, which are defined in the ensuing discussion.

The North American electric supply is subdivided in three interconnects (grids) with minimal interconnection to each other. The interconnects are:

1. Eastern Interconnect: the eastern two-thirds of the United States and Canada
2. Western Interconnect: the remainder of the United States and Canada, except most of Texas
3. Electric Reliability Council of Texas, covering most of Texas.

The reliability of the grids is managed by eight “reliability councils,” which are (grouped by interconnections):

1. Eastern Interconnect
 - a. Florida Reliability Coordinating Council
 - b. The Midwest Reliability Organization
 - c. The Northeast Power Coordinating Council
 - d. The Reliability First Corporation
 - e. The Southeast Electric Reliability Corporation
 - f. The Southwest Power Pool, Inc.
2. Electric Reliability Council of Texas
 - a. Texas Regional Entity
 - b. Western Interconnect
 - c. The Western Electricity Coordinating Council.

Each reliability council has balancing areas where load and demand are strictly monitored (for each balancing area the Area Control Error [ACE] is usually used as a metric for the imbalance between the supply and demand). The system we refer to in this document may extend over a rebalancing area or larger. The main requirement is that it shall be possible to define the electrical load that the system needs to supply and the incurred cost. Within the system it should be possible to identify several subsystems; possible subsystems include:

- Conventional baseload plants: coal, nuclear
- Renewable baseload plants: hydroelectric, geothermal, biogas, biomass, solar thermal with storage, etc.
- Intermediate load plants: hydroelectric and smaller dispatchable coal, oil-fired boiler/steam turbine plant, and natural gas-fired units (typically combined gas-turbine and steam turbine plants)
- Peaking plants like: gas turbines and gas or oil reciprocating engine power generator)
- Intermittent renewable generation, such as: wind, solar photovoltaic, concentrated solar, etc.
- Grid components, such as electrical buses, transmission lines, etc.
- Electric and thermal energy storage systems
- Energy users (process applications) such as: desalination systems chemical plants, etc.
- Ancillary service suppliers.

It should be possible also to identify the value of the following quantities:

- Load profile
- Renewable energy generation and supply
- Variable and fixed cost associated with each subsystem

- Regulatory requirements pertaining to safety and reliability

In this context, what is commonly addressed as a hybrid system or hybrid energy system will indicate a system where the activity of some of the energy users is strictly coordinated or even tightly coupled with nuclear plants, leading to larger operational flexibility of the overall system. In general, “system” will be equivalent to hybrid energy system (given the focus of this report) unless it is necessary to differentiate specifically between hybrid energy systems and conventional non-hybrid systems for clarity.

2.1 Figures of Merit

Preliminary discussion of the relevant figures of merit (FOMs) for hybrid energy systems¹ identified the following set of key FOMs that could be represented by numerical values:

1. Economic performance
2. Environmental impact, including emissions and usage of natural resources:
 - a. Greenhouses gas emission
 - b. Heat discharged to the environment
 - c. Water usage
 - d. Land usage
 - e. Efficient usage of Exergy primary feedstock content
3. Grid reliability
4. Resilience of the system with respect to exogenous perturbations (stability and recovery time of the above FOMs with variation in the price of oil, natural gas, water rights, etc.).

Assuming that the environmental impact FOM can be monetized (e.g., carbon emission tax and water rights costs) the FOMs can be reduced to three basic metrics:

- Economical performance
- System stability
- System resilience.

Monetization of system stability and system resilience is a difficult task. Capacity credit, pricing of ancillary services, capacity market, etc. are all methods that analysts and electricity market managers have tried to monetize grid stability requirements. The initial assessment of the economic benefit of hybrid energy systems is *a priori* and aims to achieve a monetization of system requirements without elaborating on market pricing strategies.

The grid must meet electricity demands. This cannot be accomplished deterministically but can only be satisfied, given the probabilistic nature of supply and demand, in a statistical sense. Probabilistic limits, imposed by the regulatory entities, segregate the frequency of events when the grid is incapable of meeting electricity demand.

The grid reliability constraint is broken into two time scales where reliability is enforced. The loss of load probability (LOLP)^{13,14} analysis is used to assess the adequacy of capacity reserves. LOLP is usually defined at the level of the interconnects; in the present case it will be used as an indicator, at the slow time scales, of the capability of the system to meet demand. At the level of the interconnect, the compensation of mismatches between supply and demand is eased by having a larger supplier base (diversification of suppliers decreases the likelihood of mismatches); therefore, acceptance criteria for LOLP is very low. In the case illustrated here the probabilistic threshold for the LOLP should be correspondingly eased, given the small size of the system considered with respect to the typical size of an interconnect and the fact that eventual mismatches could be compensated at the level of the interconnect.

Usually the most severe situation for the LOLP limit is the occurrence of the yearly peak demand at a time in which there is also a low supply of renewable generated electricity. The opposite situation (low or minimum electricity and high renewable power generation) can also present problems for balancing authorities and agent electricity generators. ACE limits are mainly challenged when an unscheduled loss of a large supplier occurs during peak demand and low supply from renewable power generators.

To evaluate hybrid energy systems here, grid stability requirements will be incorporated into the technical assessment of the integrated system by requiring that the grid supply mix is such that:

1. $LOLP < LOLP \text{ threshold}$
2. $ACE < ACE \text{ threshold}$.

Resilience can be quantified by observing the sensitivity of the system's economic performance to variation of the exogenous variable (feedstock and general resource price), although this is not an exhaustive assessment. A complete resilience analysis would require analyzing the dynamic response of the system, from the economic point of view, for a given perturbation. While this may be possible in the future, sensitivity analyses, which are first order approximations of the full resilience constraint, are initially suggested.

The problem can thus be recast as analysis (maximization) of economic performance given that:

1. LOLP test is passed
2. Grid reliability requirements are met
3. Energy market resilience is met.

2.2 The Effective Cost of Energy (ECE) as an Initial Economic Metric

Once the constraints are established for the economic analysis, the FOMs that more effectively communicate results of the economic analysis must be defined.

The first financial FOM proposed here is a slight variation on the levelized cost of electricity (LCOE). The LCOE is typically computed using the capacity factor (assumed availability rate) characteristic of the plant. In the case of hybrid systems, the proposed ECE is computed as the LCOE, but the capacity factor is replaced by the effective usage of each plant as a function of energy demand. The ECE is computed for the overall portion of the grid analyzed. ECE is also a proxy for the cost to load, but it accounts also for the capital costs.

The ECE analysis returns the average cost of producing electricity in the analyzed system (e.g., the hybrid system configuration considered). The presence of additional revenue streams other than the electricity sale needs also to be included in computation of ECE. The profit from the hybrid components—the revenue in excess of fair market value—can be folded back into the system^a with the effect of reducing the overall ECE. LCOE is often used to assess competitiveness of electricity sources in the traditional “vertically integrated” electrical power market; however, the advent of renewable power generation and deregulation of the electricity market have altered the significance, if not the calculation accuracy, of LCOE. In the present application, ECE is proposed as the consistent FOM to assess the competitiveness of the specific hybrid system configuration. ECE is superior to LCOE to the extent it is based on the effective usage of the different subsystems. ECE can be used to more effectively consider the value of energy production as a function of time, capacity additions, and statistics of intermittent energy sources and energy consumer rational decisions and demands.

The drawback of the LCOE and the related ECE analyses is that it is focused on the overall gains that could be achieved, but does not analyze in detail whether each subsystem on the grid is economically

^a The Net Present Value of the equity cash flow of the hybrid component represents a net profit in excess of fair market value when the proper discount factor is applied.

profitable. Both LCOE and ECE still hold great value in the long-term planning of energy policy, as the lowest ECE possible identifies the best possible scenario that can be achieved.

To assess the overall value of hybrid systems on the grid, ECE should be computed with and without coupling the hybrid with the grid. This analysis will determine if there is an overall positive opportunity gain in shifting from conventional generation to the introduction of a hybrid system. In this case, the analysis is very similar to that performed comparing the LCOE with the levelized avoided cost of electricity (LACE).¹⁵ Similarly, the present analysis could assess the avoided effective cost of electricity (AECE), defined as the effective cost of electricity that would be incurred within the defined system without a hybrid configuration.

2.3 Evaluation of the Economical Performance of Each Subsystem

As discussed above, a lower ECE for the hybrid system with respect to a non-hybrid configuration ensures the existence of a positive opportunity gain for the overall system but not for each subsystem. The economic feasibility of each separate subsystem needs to be analyzed to understand if, separately, each component would be economically viable. This is a necessary condition to ensure that in a free market each subsystem of the hybrid system would find the necessary financing. The incapability to finance each subsystem of the system identified by the ECE analysis would eventually impair the realization of the optimal system. Moreover, regarding the ECE analysis, the economic viability should be performed in comparative fashion, where the reference point is an identical system where the subsystems leading to the hybrid configuration (energy users) are not directly connected to the heat producers (non-hybrid configuration). Additionally, the analysis must be performed within a specific regional energy market such that the load demand can be replaced by an electricity price market.

Comparing economic performance separately for each component of the two systems (hybrid and non-hybrid) is a challenging task. Subsystems and components will likely have different life spans and capital costs. Consequently, the internal rate of return and profitability index are likely the preferred economic metrics for the comparative analysis rather than the net present value, such that one can normalize the profit to the capital investment. However, given the simplicity and better clarity of the net present value, it should also be recognized and used whenever possible.

It should be pointed out that in the ECE analysis of the hybrid system, each subsystem is seen as a price maker, given that the ECE is the price of electricity that ensures a fair profit for the electricity supplier. Conversely, in the evaluation of the profitability of each subsystem, the system is a price taker with respect the grid. It is worth observing that overall, an ECE for the hybrid subsystem that is lower than the AECE ensures the existence of a pricing strategy that will allow the overall hybrid system to have a positive NPV when seen as a price taker.

2.4 Economic Analysis to Assess the Natural Evolution of the System

The last step of the economic analysis is to understand what natural market forces would cause grid to add subsystems with a hybrid configuration. This evolution would be driven by the economic assessment of each single component on the grid to decide which new components should be added. Unfortunately, no assurance can be made that the grid will evolve to the lowest ECE since the evolution is mostly driven by short and mid-term economic considerations that could result in different choices than what would lead to the best long-term scenario (minimum ECE).

2.5 Summary of the Performance Analysis Approach for Hybrid Systems

The performance of a hybrid system can be assessed using the following steps:

1. Find the hybrid system configurations that minimize ECE under the defined constraints (LOLP, stability, and resilience)

2. If the ECE of the hybrid system is lower than the AECE of the conventional system, then analyze the profitability of each individual subsystem that comprises the integrated hybrid system. Possibly redistribute the profit to assure profitability of each subsystem.
3. Analyze the natural evolution of the grid under market forces to evaluate the likelihood of an evolution toward the hybrid configuration identified in the 2014 Idaho National Laboratory (INL) report (INL/EXT-14-32857).¹

3. DEFINITION OF SIMULATION NEEDS

As discussed in Section 2, several steps must be performed to analyze the successful deployment of hybrid systems in a specified region/market. The steps defined in Subsection 2.5 can be reformulated in a fashion that is more prone to a mathematical representation that can later be used to establish the needs for specific models and algorithms.

- Solve a constrained optimization problem to minimize the ECE objective function (the ECE must be less than the AECE of the conventional grid). The boundary conditions are defined by the following parameters, each of which may vary with time:
 - Electricity demand
 - Resource price (cost of fuels and feedstock for the hybrid systems)
 - Renewable energy supply
 - Price of the hybrid component products (e.g., hydrogen, synfuel, clean water).
- The problem is constrained by:
 - Sensitivity to exogenous variables (i.e., boundary conditions)
 - Stability of the grid (the exact formulation of this constraint is provided in the next section).
- Achieve an improved positive economic performance (net present value, internal rate of return, and profitability index) with respect to the non-hybrid system for subsystems individually connected to the grid (using the system configuration solution of the previous step with the following boundary conditions:
 - Electricity price
 - Resource price (cost of fuels and feedstock for the subsystems in the hybrid configuration)
 - Renewable energy supply
 - Price of the hybrid component products (e.g., hydrogen, synfuel, clean water).
- Analyze the evolution of the electricity supply mix under the market forces over time using the profitability metrics derived in the previous step for the grid components.

3.1 ECE Analysis

3.1.1 Software Requirements

The final goal of this analysis is to establish the energy supply mix that minimizes an overall objective function. The objective function should incorporate the leading FOMs that were identified in the foundational workshop report¹ and summarized in Subsection 2.1.

In the suggested analysis approach the objective function to be minimized could be based on the ECE under the following constraints:

- Limited oscillation of the price of electricity and hybrid energy products or services under exogenous oscillation of feedstock costs
- Grid stability satisfying the regulatory prescription.

The grid stability constraint translates to two different requirements at two different time scales. The LOLP is usually computed by testing the capability to meet the electricity demand on an hourly basis. At the same time, the grid is tested over a much shorter time scale (10^{-5} sec to 10 min) to verify that the inertia of the grid and the spinning reserves are sufficient to absorb possible instabilities over a range of 10 minutes to 1 hour. These tests are also performed to verify that the spinning and fast responding reserves can properly react to the loss of generation following the trip of the largest supplier present on the grid. These two stability requirements are carried out separately; the computational burden to perform these statistical tests jointly would be overwhelming. Figure 1 shows the different time scales at which different resources are used.

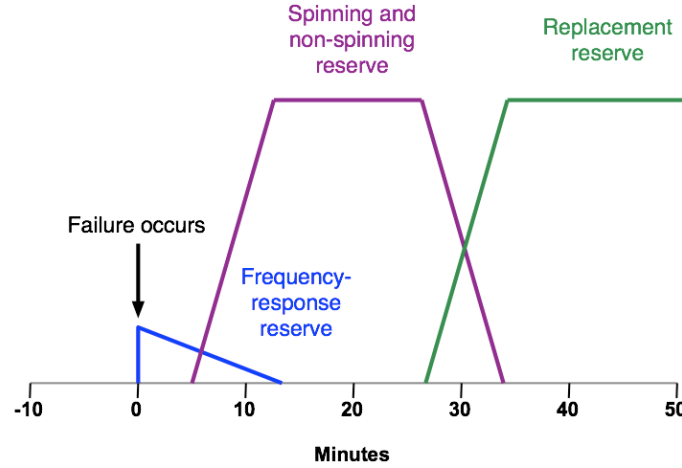


Figure 1. Time scales for the specified stabilizing energy reserves. Several approaches work to assess the different grid reliability requirements.^{14,16} Generally, the most-accurate and advanced methodologies use a method of statistical analysis of the time series of the load and renewable supplies (e.g., wind and solar).

The constraint established, based on the resilience of the electricity price, can be checked by testing the impact of feedstock price oscillation on the final ECE. More theoretical work will be necessary to define proper quantitative metrics in this area.

The simulation schema can be summarized as follows (see Figure 2):

1. Establish a set of optimization parameters (ranging from the fraction of energy supplied by each type of energy generation source to control parameters impacting operation management of hybrid systems)
2. Define an initial set of values for the control parameters
 - a. For each set of parameters:
 - (1) Perform the stability checks
 - (a) Select representative time intervals (up to 10 minutes with 1-minute resolution)
 - (b) Perform a statistical analysis of the failure rate
 - (2) If the stability check is passed, perform the LOLP test
 - (a) Select representative time intervals (up to 1 week with 1-hour resolution)
 - (b) Perform a statistical analysis of the failure rate
 - (3) If the LOLP test is passed, calculate the ECE
 - (a) Use a sampling strategy to select prototypic days for a 1-year cycle

- (b) Compute ECE, tracking other possibly useful data (e.g., greenhouse gas emissions)
- b. Select a new set of control parameters to seek the minimum of the ECE and restart from Point 2.a. described above. Depending on the degree of uncertainty in the ECE, which is driven by the randomness of the yearly time histories of electricity demand and renewable generation, the problem should be cast as a constrained optimization or a constrained optimization under uncertainty.
- c. Compute the sensitivity coefficients (using the outcome of several simulations) and test economic resilience of the energy supply portfolio.

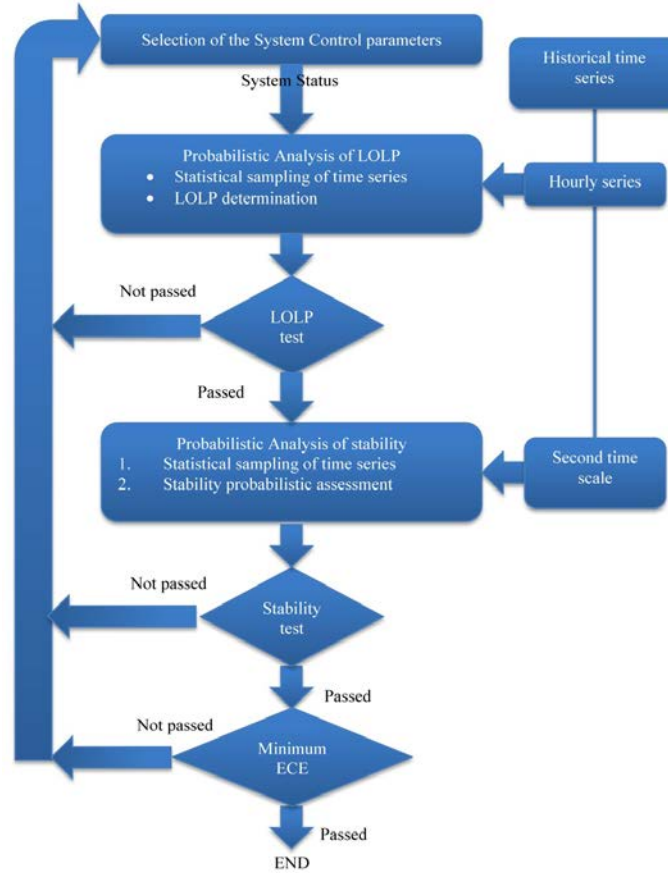


Figure 2. Simulation flow.

It is currently unclear if the stability test should be performed inside the optimization loop or if it can be brought outside, as an external confirmatory test on the final optimization results, or if a hybrid approach should be used. This choice will depend on how much the possible adjustments necessary to pass the grid reliability test will impact the defined FOMs. If this impact is negligible then Step 2.a.1 can be moved outside the optimization loop, resulting in a significant computational savings. Unfortunately, the same strategy appears to be infeasible for the test on the LOLP, as its outcome will directly impact the admissible fraction of renewable penetration.

3.1.1 Necessary Algorithms

The algorithm needs are highlighted in Table 1, based on the calculation flow summarized in Subsection 3.1.1.

Table 1. Correspondence between tasks, corresponding algorithm type and current development status.

Task	Algorithm Type	State of Knowledge
Generate the time series to perform the statistical analysis for the LOLP and the stability analysis maximizing relevance to the analysis performed.	Several approaches are possible: <ul style="list-style-type: none"> • Sampling strategies of the time series • Statistical analysis of the time series to prototypical time series • Statistical analysis of the time series for modal analysis of the system reliability distribution. 	Time series analysis is a very active research field. Open source libraries are available, such as Pandas. ¹⁷ Research is necessary to identify algorithms to select high-risk (high probability and high consequence) time series samples. Sampling strategies could be implemented in the RAVEN code ⁴ (discussed in Subsection 4.3 as a possible candidate for the general driver of the simulation).
Forecast of demand and renewable supply play a great role in the reliability analysis of the grid. For this reason forecasting algorithms need to be considered as part of the analysis framework.	Two possible option have been considered: <ul style="list-style-type: none"> • For some markets electricity demand forecasts are available (e.g., California Independent System Operator) that could be sampled as historical time series • Prediction algorithms used by the grid operators.¹⁸ 	The grid is exposed to significant risk given mismatch in the forecast and the actual market situation; hence, this is a highly active research field. Auto Regressive Moving Average or similar algorithms ¹⁸ could be employed using open sources libraries (e.g., Pandas ¹⁷).
Derive failure rate probability over long periods of time from the above sampling strategies.	Inference of statistical properties of complex system is tightly coupled with the sampling strategy used. Usually a mixture of unsupervised learning, supervised learning, and active supervised learning algorithms are used.	Algorithm development and choice will be a direct consequence of the sampling strategies and the statistical weights assigned to the samples. RAVEN currently provides a large set of supervised learning algorithms and is developing new unsupervised learning and active supervised learning algorithms.
Compute the ECE and greenhouse gas emissions.	Financial algebraic formulas	This is a rather straightforward step using classical financial management formulas and information derived from the grid model.
Perform constrained optimization under uncertainty (minimization of ECE under uncertainties of renewable supply and demand).	Mixed integer and continuous optimization with probabilistic constraints	There are several available open source optimization libraries (e.g., Scipy, Optimize). In the current application it will be necessary to evaluate the most suitable choice while considering that the number of optimization parameters could be fairly large (~100) and each single run will be fairly expensive. A parallel optimization framework may be necessary. Optimization under uncertainty will require algorithms available in standard libraries to be improved.
Compute sensitivity coefficients for the figures of merit with respect the exogenous variables	Multi-dimensional linear regression for continuous and integer variables	Algorithms to compute sensitivity coefficients are fairly well known. Basic linear regression algorithms will be sufficient and could be directly applied to the database that will be generated by the optimization search. These types of algorithms are already integrated in the RAVEN code, as will be discussed in Subsection 4.3.

A large number of runs will be necessary to obtain a meaningful statistical estimation of grid failure rates while simultaneously optimizing system operation. One possible option to overcome this challenge is to use surrogate models to accelerate the convergence rate. Surrogate models are better known in the statistical analysis field as supervised learning algorithms. Those types of algorithms are widely used in optimization and uncertainty quantification fields to speed up the simulation time as illustrated in Queipo et al.¹⁹ The current problem is an area in which this application field is not completely mature. A flow chart for the possible utilization of a reduced order model for optimization is shown in Figure 3.

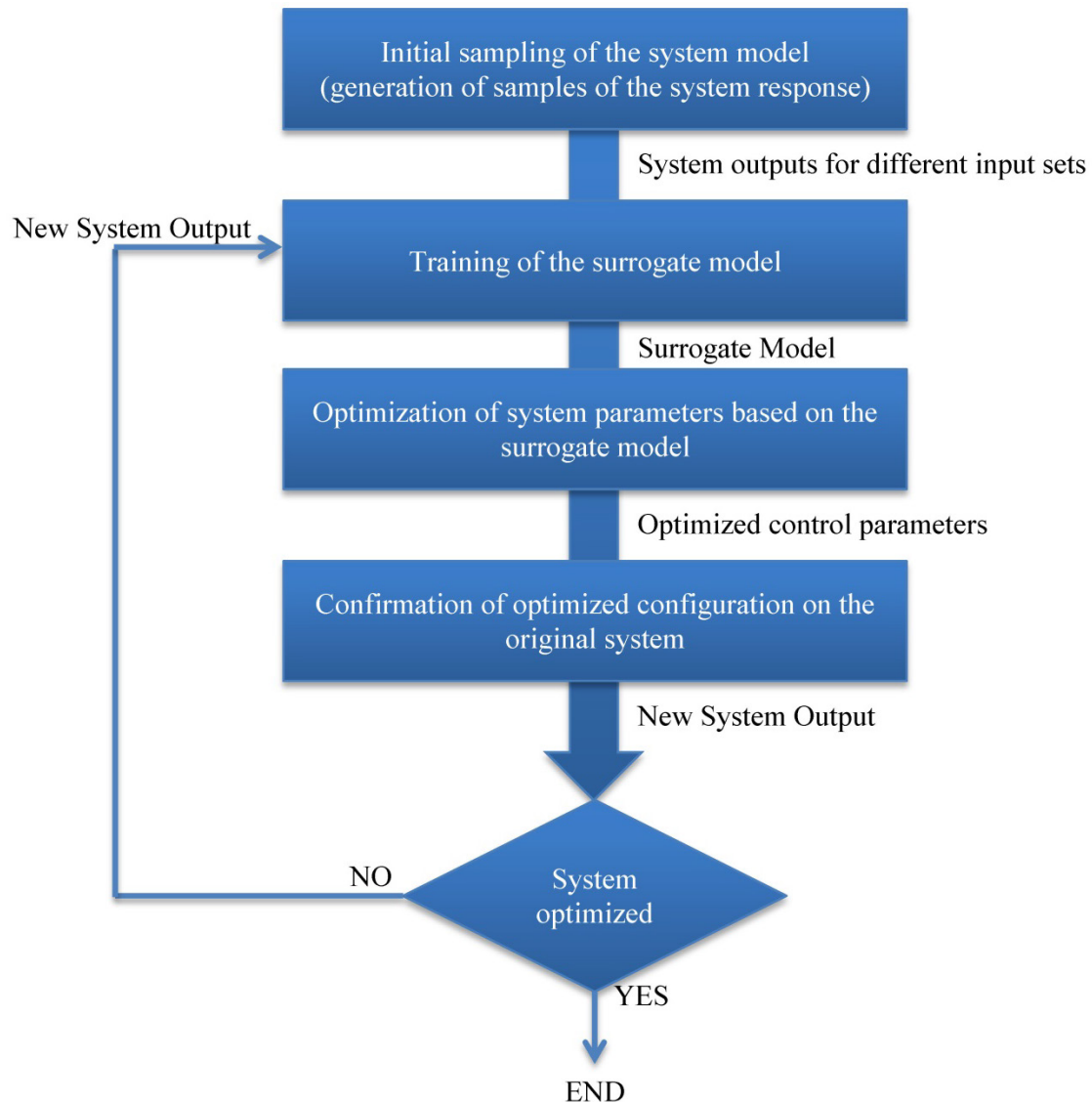


Figure 3. Possible utilization of a reduced order model for optimization.

A software package that is capable of implementing both the grid reliability analysis and the optimization search for a complex system using surrogate models has not been identified. This is likely the largest gap in the comprehensive analysis of integrated hybrid systems on the grid.

3.1.2 Necessary Grid Component Models

The basic algorithms needed to perform what has been defined as ECE analysis are described in the previous sections. At this point it is necessary to more clearly identify the necessary grid models. The grid models will serve three purposes:

1. Provide a grid model to test grid stability under a loss of generation capacity scenario (high-frequency analysis)
2. Provide a grid model to determine the LOLP
3. Determine the commodity and cash flows to evaluate the ECE, greenhouse gas emissions, and sensitivity to exogenous perturbations.

Given the time scale at which the LOLP test will be performed (hourly resolution over weekly time periods) it is foreseeable that the same grid model could be used for Items 2 and 3. This grid model would include numerical models of grid subsystems at a level of resolution (accuracy) that corresponds to the error already present in the source term (hourly average of the energy demand and renewable supply). For this analysis the source term is, in fact, approximated by hourly averages; therefore, it is unnecessary to model the turbine shaft inertia. However, the models will have to accurately represent the thermo-mechanics of system components and operational controls.

It will be necessary to run the models on large computing clusters given the large number of simulations that calculating the LOLP and system optimization will require. Therefore, these models should be built in languages or meta-languages that support such an option. Several models are already available, but unfortunately, are written in different languages or meta-languages. A proper integration platform capable of managing the information exchange among the different component models must be identified.

All the modeled subsystems/components need to communicate with each other and be synchronized with the time integration schemes. This can be achieved by using a master-slave infrastructure (see Figure 4). In the complex case considered here, this situation can be further complicated by the different platforms and languages used by the existing, accepted models across diverse fields (e.g., nuclear processes, chemical processes, electric infrastructure). It is necessary to establish a common interface that allows the communication between the different subsystem models.

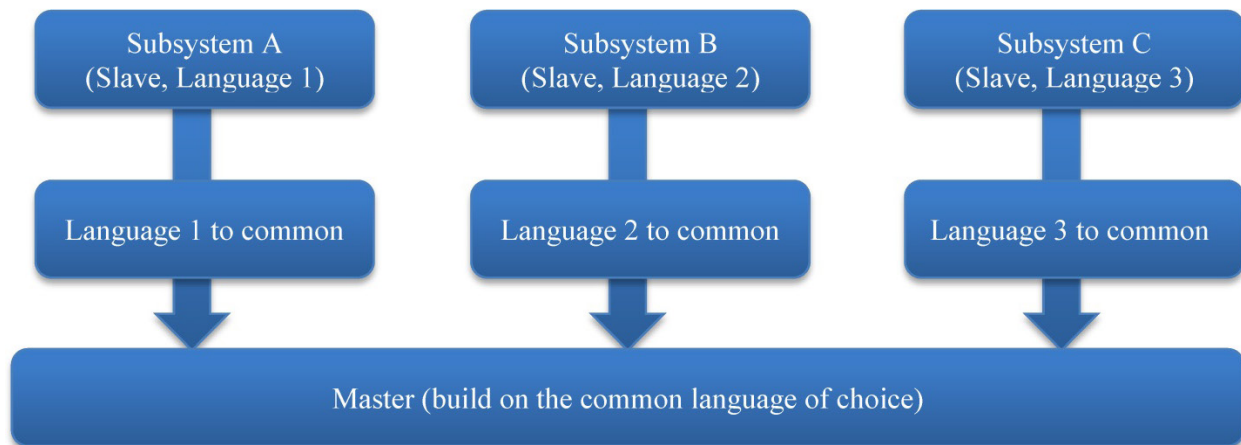


Figure 4. Slave-master code structure in presence of multiple types of language.

Simulation of the grid to perform the stability test is further challenged by the very small time scales of physical phenomena (e.g., 10^{-5}) that must be considered for transients lasting for tens of minute. A reasonable simulation speed may be achieved by using dedicated hardware, such as RTDS, to simulate grid components. Unfortunately, the availability of those hardware components is limited. If possible, use of models developed on standard hardware platforms (i.e., clusters) is preferred, if a satisfactory computational speed can be achieved. While the latter approach reduces the use of larger computational resources, it poses the challenge in coupling subsystem models running on dedicated hardware with subsystem models running on standard clusters. The choice to explore such coupling will depend on the trade-off between acquiring more dedicated hardware and investing in the integration effort under the constraint that, at the end, the entire local portion of the grid should be simulated.

Realistic system simulation requires accessing time history databases, specifically the time history of the renewable supply and energy demand. The time-dependent data will be needed on both an hourly basis (for the LOLP test) and on a second basis (for the reliability test).

The above discussion can be summarized with the following requirements:

- Models to determine satisfaction of the LOLP test:
 - Mixed languages/meta-languages require an integration platform
 - Models should be compatible with large computing clusters
 - The thermo-mechanics representation of the system must be accurate
 - Relevant time scales are a few minutes to days
 - Simulation should be carried out over a 1-year period.
- Models to determine grid reliability:
 - Models should be built on dedicated software.
 - If necessary, some additional models may be built in mixed languages/meta-languages. In this case an integrating platform will be necessary.
 - Models built on non-dedicated hardware must be compatible with large clusters.
 - Phenomena affecting the grid at time scales between 10^{-5} s and few minutes must be accurately represented.
 - Simulation should be carried out over a 10-minute period.
- Time histories, required on both an hourly and minute basis
 - Energy demand
 - Renewable availability.

3.2 Profitability Analysis of Grid Components

Once the tools to perform the LCOE analysis have been selected/defined, the profitability analysis of each grid component is relatively straightforward. The component modeling needs, from the physical point of view, are the same as those required to establish the cash and commodities flow in the LCOE analysis. As discussed in Subsection 3.1, those models are also used for the LOLP constraint test. Consequently, when a component is highlighted as available in the “availability” column in the section “LOLP time scale” in Appendix A, it should be considered available for the profitability analysis of the individual grid component.

In the profitability analysis the system will act as a price taker, unlike the LCOE analysis in which the system acts as a price maker (more exactly, only costs are traced). Hence, the simulation should be performed using the same optimized system derived in the LCOE step, but it should be designed to react to variation in the electricity price. The time series for the flow of commodities from the simulation results will need to be mapped to an equivalent cash flow analysis (an example of how this could be done is illustrated in the 2014 INL report¹). The cash flow for each component will then be used to derive the financial FOMs for each component.

In summary, the following developments will be necessary in addition to the infrastructure already in place for the LCOE analysis:

1. Adaptation of the system control to respond to electricity price rather than electricity demand.
2. An application capable of transforming the network of commodity flows into the corresponding cash flow.
3. The time histories of the electricity price is required. It is anticipated that data having a 15-minute resolution will be appropriate, although this may depend on the characteristics of the electricity spot market.

3.3 Evolution Analysis

Evolutional analysis focuses on predicting how the energy supply mix may change under reasonable market forces. The objective is to analyze the possible evolution of a large conventional grid toward the adoption of subsystems with a hybrid configuration (e.g., nuclear plant plus heat user). Even if the objective function used to determine the market pressure (cost minimization) fully corresponds to the objective function used in the ECE analysis, the final energy mix may differ from the one identified as ideal in the static analysis. This results from the possible existence of local extremes in the cost function that make it impossible to reach the global maximum/minimum. One should note that the evolution analysis will lead to economic performance that is, at best, equal to the best scenario identified by the ECE analysis.

The simulation environment should be able to reproduce the following evolutionary scenario:

1. A forecast of the generation units that will be shut down and the increase/decrease in electricity demand will generate the electricity supply gap that must be filled.
2. Several electricity generation options compete based on financial metrics for a given electricity market, under grid reliability constraints. The reliability of the grid can be accounted for either by a direct test of the LOLP or by using approximate formulas to compute the credit value,^{20,21} which is an approximate way to embed reliability requirements in the economic evaluation of the electricity supplier.
3. As the supply mix changes, the simulation of the electricity market must be restarted with Item 1 to account for new demand, supply, and different credit value.

This type of simulation is challenged by the need to estimate the effective utilization of each supplier that depends on the energy market pricing strategy, while maintaining the reliability constraint. As previously discussed for the LOLP, approximating methodologies exist and it is possible to derive reasonable approximation to determine the effective utilization of each supplier.

4. SIMULATION FRAMEWORK

This section defines a possible simulation framework for hybrid energy systems and identifies currently known gaps in achieving this simulation strategy. The potential software languages, modeling platforms, currently available software, and infrastructure are introduced with additional details on their capabilities.

First, the framework and tools required to perform the LOLP and grid reliability analysis must be differentiated. The analysis carried out here assumes that the reliability of the grid has a negligible impact on the overall ECE once LOLP criteria are satisfied. If this assumption is true, then the models needed for the LOLP and the reliability analyses do not need to interact. At a minimum, the reliability test could be performed periodically with respect the LOLP test.

ECE are best calculated for the system with an integrated model that simulates the coordinated dynamic transfer of energy, energy conversion, and products of energy services by (1) traditional energy service suppliers, and (2) the integrated energy users. This requires as a minimum a computational representation of the thermal, mechanical, chemical, and electrical processes in the systems as process units, reactors, manufacturing plants, and energy delivery (to the appropriate point of interface with the market transaction, such as an electricity bus, or a product depot or distribution terminal where a commodity price is established). The modeling platform for the integrated systems can address systems controls and operational coordination with respect to functional (technical) and operating (economical) objectives. This system model is required to establish capital costs, operating and maintenance costs, instruments and monitors for controls, and systems diagnostics and prognostic/supervisory or automatic controls. Such detail is essential to calculating economic pro forma. Additionally, the hybrid systems models can be used to forecast feasible operating limits and conditions. Computational efficiency is provided by the integrated process models before detailed analysis with real-time digital and real-time hardware, grid, and control inputs in co-simulation and process development environment. A detailed discussion on developing hybrid process models is provided in a companion report.²²

It is foreseeable that the reliability analysis will also utilize models running on RTDS hardware when the time scale of operational dynamics is needed to address LOLP and ACE. Simplified plant models, if needed, can be generated in PSCAD. The challenge in using such a configuration is the large number of simulations that will be necessary for the statistical analysis. This number will need to be spread over a sizable computing cluster, but the number of RTDS units is constrained. To overcome this difficulty a sizable amount of work should be devoted to:

1. Limiting the number of simulations needed to assess reliability
2. Keeping the plant models as simple as possible with regard to physical phenomena that have low impact on the fast time scales at which grid reliability is assessed
3. Simultaneously using RTDS resources that are geographically distributed (possibly located at different laboratories).

RAVEN is a possible candidate for statistical pre- and post-processing. More detail on the software and its development status is provided in Subsection 4.3; at this stage it is useful to simply present the idea and contextualize the schema described above and summarized in Figure 5.

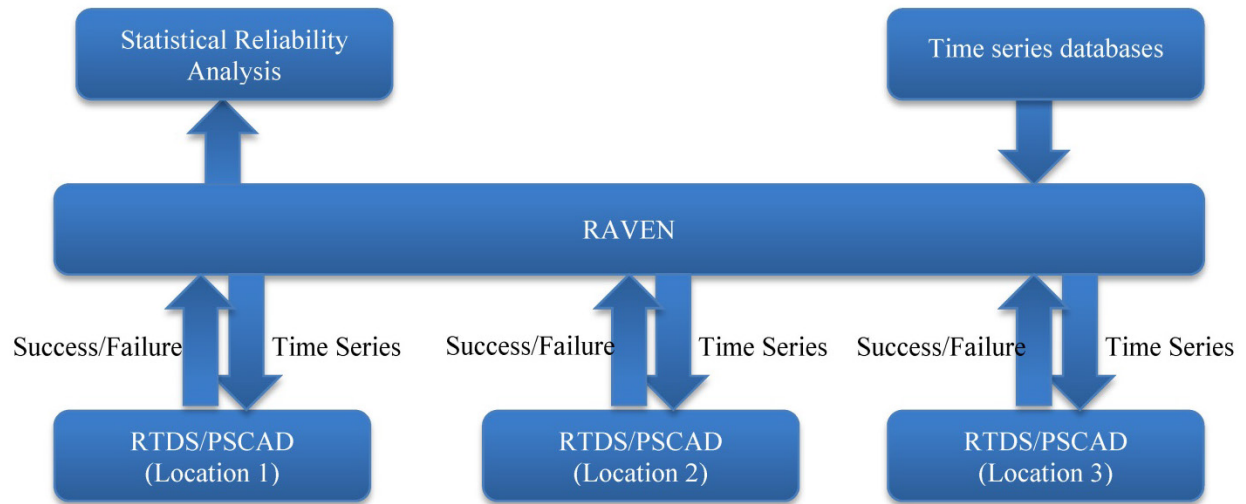


Figure 5. Possible configuration where RAVEN is used as a driver for a statistical analysis using a distributed system of RTDS.

For the ECE and LOLP the modeling and simulation landscape is more fragmented. Several component models are already available across various platforms/languages. For this reason, it is necessary to introduce the equivalent of a global translator (see Figure 6). A candidate translator is the protocol described by the Functional Mockup Interfaces and Functional Mockup Interfaces (FMI) Functional Mockup Unit (FMI/FMU) protocol.²³ All the languages and development platforms utilized in the performance analysis of integrated hybrid energy systems will be required to generate a binary code (FMU). The interface of these FMUs is described using FMI standards. More details on this process are provided in Subsection 4.1. At this point it is necessary to simply highlight the requirement for each model used in the LCOE analysis to support a translation into FMI/FMU.

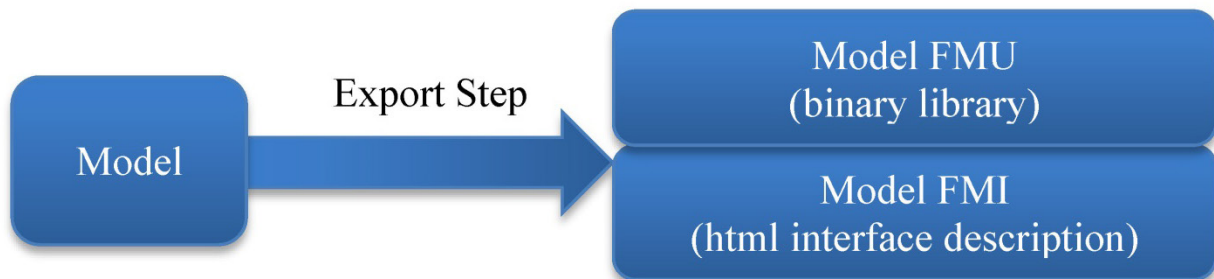


Figure 6. Translation of a generic model to be used under the FMI/FMU protocol.

Once translated into FMUs, the models still need to be coordinated such that they progress synchronously over time and represent the specific grid control logic. This task could be implemented in the PTOLEMY framework.²⁴ PTOLEMY, as discussed in Subsection 4.2, is capable of using the FMI descriptions of the individual models to build the proper communication network. Additionally, PTOLEMY can act as a driver (master) that is capable of evolving the simulation over time, or the current application to the simulation of hybrid energy systems, the control logic implementation will be minimized at the global level (PTOLEMY level), moving as much of the control logic as possible into the individual subsystem representations.

The size of the computational effort requires a large computational cluster, as several thousand PTOLEMY runs (embedding the models FMUs) will be needed. The compatibility of any software considered for this application is mandatory.

As mentioned, it will be a large task to generate the series sampling, the statistical post processing, and the optimized operative settings. Currently, other Department of Energy (DOE)-funded programs (e.g., Nuclear Energy Advanced Modeling and Simulation and Light Water Reactor Sustainability) have expressed similar needs and are currently supporting the development of RAVEN, which is a likely candidate to fulfill such tasks. RAVEN could serve as both pre- and post-statistical processor (i.e., generating the time series and estimating the LOLP) and the LCOE optimization engine.

Before assessing the different software platforms and languages, it is necessary to mention that while most of the other subsystems will be assembled/developed/implemented specifically for this program, the nuclear plant would be modeled using a specific software package (RELAP5-3D) developed exactly for that purpose, which is presented in Section 4.6.

4.1 Functional Mockup Interfaces and Functional Mockup Unit

FMI and FMU are tools of independent standards (communication tools employing an independent standard for communication) to allow run time communication between software programs. The FMI/FMU concept has evolved to communicate among numerical models of physical components built using different languages and platforms. A component model built using a generic platform or language needs to be wrapped using a C-binding or directly translated to C. The C wrapper or the translated C code should be capable of exchanging information with the simulation environment following a general protocol (template). The code generated in this manner is called an FMU, while the accompanying xml files that describe the interface-specific implementation in detail (e.g., variable names, units, time stepping etc.) are called FMIs. An FMU that is not originally written in C may be generated two ways—direct translation to C or construction of C binding.

4.1.1 Construction of C-Binding Interfaces

Specific language-to-language interfacing capabilities exist, such as FORTRAN to C, that would allow implementing C binding, but interfaces are not available for all possible languages that could be used to generate the numerical models relevant to hybrid systems. BABEL is a useful tool for accomplishing this task.²⁵ BABEL has the advantage of being a long-lasting project that is actively developed. Figure 7, taken from the BABEL website, illustrates the chain of interfaces possible using BABEL as a global translator. For example, BABEL could be used to construct interfaces toward FMI/FMU for models not built in C, but in any of the other languages with which BABEL is capable of communicating.

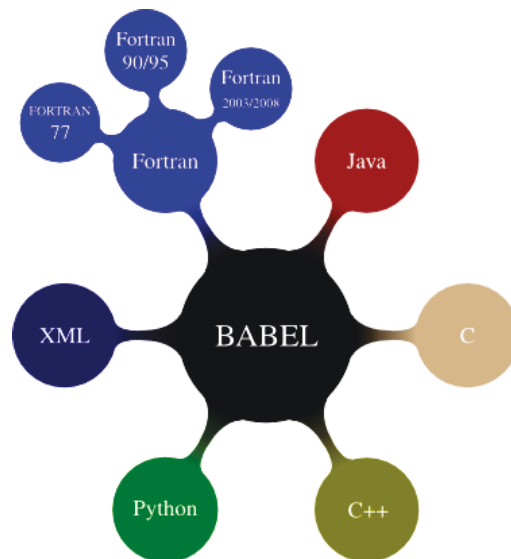


Figure 7. Graphical representation of cross-linking capabilities of the BABEL library (source: <https://computation.llnl.gov/casc/components/index.html#page=home>).

4.1.1 Platform with Direct Interface to FMU

Table 2, taken from the official FMI/FMU website (<https://www.fmi-standard.org>) provides information on the software platforms currently capable of interacting directly with FMI/FMU. Definition of the import/export and the slave/master designations for the model exchange and co-simulation environments, respectively, are as follows:

- Model exchange:
 - Import: the software is capable of importing and using models passed as FMI/FMU
 - Export: the software is capable of exporting models internally developed as FMI/FMU.
- Co-simulation:
 - Slave: models developed using this software can be driven by FMI/FMU
 - Master: models developed using this software can drive models passed as FMI/FMU.

Table 2. Compatibility table for the Functional Mockup Unit (FMU) and Functional Mockup Interfaces (FMI) protocols.

Tool Supporting FMI	FMI Version	Model Exchange		Co-simulation		Notes
		Export	Import	Slave	Master	
Adams	FMI_1.0		Planned	Available	Available	High end multibody dynamics simulation software from MSC Software
Amesim	FMI_1.0	Available	Available	Available	Available	Integrated simulation platform for the analysis of multi-domain mechatronics systems by Siemens PLM Software
ANSYS SCADE Display	FMI_1.0	Available		Available		SCADE Display facilitates embedded graphics, display and HMI development and certified code generation for safety-critical displays from ANSYS .
ANSYS SCADE Suite	FMI_1.0	Available		Available		SCADE Suite is a model-based development environment with certified code generation for safety critical embedded applications from ANSYS .
ANSYS Simplorer	FMI_1.0		Available	Planned		ANSYS Simplorer is a multi-domain, multi-technology simulation program from ANSYS .
ASim - AUTOSAR Simulation	FMI_1.0	Available		Available		AUTOSAR product from Dassault Systèmes
@Source	FMI_1.0	Available				Simulink via @Source
AVL CRUISE	FMI_1.0	Planned	Available	Available	Available	Vehicle system analysis tool for the optimization of fuel efficiency, emission, performance and drivability, from office to HiL to test bed.
Building Controls Virtual Test Bed	FMI_1.0				Available	BCVTB is a Software environment, based on Ptolemy II , for co-simulation of, and data exchange with, building energy and control systems.
CarMaker	FMI_1.0				Available	CarMaker is an open test and integration platform for MiL, SiL, and HiL.
CATIA	FMI_1.0	Available	Available	Available	Available	Environment for Product Design and Innovation, including systems engineering tools based on Modelica, by Dassault Systèmes
ControlBuild	FMI_1.0	Available	Available	Available	Available	Environment for IEC 61131-3 control applications from Dassault Systèmes
CosiMate	FMI_1.0		Available		Available	Co-simulation Environment from ChiasTek

Table 2. (continued).

Tool Supporting FMI	FMI Version	Model Exchange		Co-simulation		Notes
		Export	Import	Slave	Master	
Cybernetica CENIT	FMI_1.0		Available		Planned	Industrial product for nonlinear Model Predictive Control (NMPC) from Cybernetica.
Cybernetica ModelFit	FMI_1.0		Available		Available	Software for model verification, state and parameter estimation, using logged process data. By Cybernetica.
DSHplus	FMI_1.0	Planned		Available		Fluid power simulation software from FLUIDON .
dSPACE SCALEXIO	FMI_1.0				Available	dSPACE SCALEXIO is a Hardware-in-the-Loop (HIL) integration and simulation platform from dSPACE . Please also refer to the dSPACE FMI sites for more information about the FMI 1.0 and FMI 2.0 support.
dSPACE SYNECT	FMI_1.0				Planned	dSPACE SYNECT is a data management tool from dSPACE that enables you to manage FMUs and Simulink models as well as their dependencies, versions, and variants throughout the entire software development process. Please also refer to the dSPACE FMI sites for more information about the FMI support.
dSPACE VEOS	FMI_1.0				Available	dSPACE VEOS is a PC-based virtual integration and simulation platform from dSPACE . Please also refer to the dSPACE FMI sites for more information about the FMI 1.0 and FMI 2.0 support.
Dymola	FMI_1.0	Available	Available	Available	Available	Modelica environment from Dassault Systèmes. ModelExchange also available for Simulink using Simulink Coder.
EnergyPlus	FMI_1.0			Available	Available	Whole building energy simulation program.
ETAS - ASCMO	FMI_1.0			Available		Creation and export of statistical (meta) models using Design of Experiments (DoE) from ETAS .
ETAS - FMI-based Integration and Simulation Platform	FMI_1.0		Planned		Planned	Integration and simulation platform based on FMI 1.0 from ETAS .

Table 2. (continued).

Tool Supporting FMI	FMI Version	Model Exchange		Co-simulation		Notes
		Export	Import	Slave	Master	
ETAS - FMU Generator for ASCET	FMI_1.0	Planned				FMU Generator for ASCET from ETAS .
ETAS - FMU Generator for Simulink®	FMI_1.0	Planned		Planned		FMU Generator for Simulink® from ETAS .
ETAS - INCA-FLOW (MiL/SiL Connector)	FMI_1.0		Planned		Available	Guided and automated calibration of FMUs with connection to ETAS INCA.
ETAS - ISOLAR-EVE (ETAS Virtual ECU)	FMI_1.0			Available		PC-based platform from ETAS for ECU software validation at the component, subsystem or system level; allows for validation of Application Software, Basis Software and complete ECU software in a virtual environment.
ETAS - LABCAR-OPERATOR	FMI_1.0				Available	Frontend for ETAS HiL systems LABCAR, operating on the creation of experiments and their subsequent execution.
Flowmaster	FMI_1.0	Available				High capability 1D CFD tool for thermo-fluid simulation from Mentor Graphics .
FMI Add-in for Excel	FMI_1.0				Available	FMI Add-in for Microsoft Excel by Modelon . Offers support for batch simulation of FMUs.
FMI add-on for NI VeriStand	FMI_1.0		Available		Available	NI VeriStand supports FMI through the use of the FMI add-on for NI VeriStand from Dofware .
FMI Blockset for Simulink	FMI_1.0				Available	Import of FMI Co-Simulation models into Simulink - provided by Claytex .
FMI Library	FMI_1.0		Available		Available	Open source (BSD) C library for integration of FMI technology in custom applications by Modelon .
FMI Target for Simulink Coder	FMI_1.0			Available		Export of stand-alone FMUs for Co-Simulation from Simulink using Simulink Coder - provided by ITI .
FMI Toolbox for CarMaker	FMI_1.0		Available		Available	For IPG CarMaker via FMI Toolbox for CARMAKER from Modelon .

Table 2. (continued).

Tool Supporting FMI	FMI Version	Model Exchange		Co-simulation		Notes
		Export	Import	Slave	Master	
FMI Toolbox for MATLAB/Simulink	FMI_1.0	Available	Available	Available	Available	The FMI Toolbox for MATLAB/Simulink from Modelon enables FMU import and export for MATLAB/Simulink for both model exchange and co-simulation.
FMUSDK	FMI_1.0	Available	Available	Available	Available	FMU Software Development Kit from QTronic .
GT-SUITE	FMI_1.0		Available	Available	Available	Multi-Physics Simulation Platform for Powertrain and Vehicle Systems.
Hopsan	FMI_1.0	Available	Available			Hopsan is a free simulation tool developed at Linköping University. It is using distributed solver techniques with good support for parallelism using multi-core processors.
IBM Rational Rhapsody	FMI_1.0	Available	Planned	Planned	Planned	IBM® Rational® Rhapsody® family provides a collaborative design, development, and test environment for systems engineers and software engineers.
ICOS “Independent Co-Simulation”	FMI_1.0		Available	Available	Available	ICOS is a co-simulation tool developed by Virtual Vehicle .
JavaFMI	FMI_1.0		Planned		Available	JavaFMI is a Java wrapper for the Functional Mock-up Interface (V1.0 or V2.0) both for Windows 32 & 64 bit OS.
JFMI	FMI_1.0			Available	Available	A Java Wrapper for the Functional Mock-up Interface, based on FMU SDK.
JModelica.org	FMI_1.0	Available	Available	Available	Available	Open source Modelica environment from Modelon .
LMS Virtual.Lab Motion	FMI_1.0		Available	Available	Available	Virtual.Lab Motion is a high end multi body software from LMS International
MapleSim	FMI_1.0	Available	Planned	Planned	Planned	Modelica-based modeling and simulation tool from Maplesoft.
MESSINA	FMI_1.0		Available		Available	MESSINA is a test platform for model-based ECU function development.
MWorks	FMI_1.0	Available	Planned	Planned	Planned	Modelica environment from Suzhou Tongyuan.
NI LabVIEW	FMI_1.0		Planned			Graphical programming environment for measurement, test, and control systems from National Instruments.
OpenModelica	FMI_1.0	Available	Available	Planned	Available	Open source Modelica environment from OSMC.
OPTIMICA Studio	FMI_1.0	Available	Available	Planned	Available	Modelica environment from Modelon.

Table 2. (continued).

Tool Supporting FMI	FMI Version	Model Exchange		Co-simulation		Notes
		Export	Import	Slave	Master	
Ptolemy II	FMI_1.0				Planned	Software environment for design and analysis of heterogeneous systems.
PyFMI	FMI_1.0		Available		Available	For Python via the open source package PyFMI from Modelon . Also available as part of the JModelica.org platform.
RecurDyn	FMI_1.0	Planned	Planned	Planned	Available	High End Multi Flexible Body Dynamics Software from FunctionBay .
Reference FMUs	FMI_1.0	Planned		Planned		Reference FMUs supplied by enthusiasts and volunteers to show case specific FMU features.
Silver	FMI_1.0	Available	Available	Available	Available	Virtual integration platform for Software in the Loop from QTronic .
SIMPACK	FMI_1.0	Planned	Available	Available	Available	High-end multi-body simulation software from SIMPACK AG
SimulationX	FMI_1.0	Available	Available	Available	Available	Multi-domain simulation tool for design, analysis, and virtual prototyping of complex systems by ITI .
SystemModeler	FMI_1.0	Available	Planned	Planned	Planned	Modelica environment from Wolfram Research .
TLK FMI Suite	FMI_1.0		Available		Available	TLK FMI Suite provides LabVIEW and Simulink blocks for FMU simulation.
TLK TISC Suite	FMI_1.0		Available		Available	Co-simulation environment from TLK-Thermo.
TWT Co-Simulation Framework	FMI_1.0			Available	Available	Communication layer tool to flexibly plug together models for performing a co-simulation; front-end for set-up, monitoring and post-processing included.
TWT FMU Trust Centre	FMI_1.0			Available		Cryptographic protection and signature of models including their safe PLM storage; secure authentication and authorization for protected (co-)simulation.
xMOD	FMI_1.0		Available		Available	Heterogeneous model integration environment and virtual instrumentation and experimentation laboratory from IFPEN distributed by D2T .

4.2 Ptolemy

Ptolemy II (<http://ptolemy.eecs.berkeley.edu/ptolemyII/>) is an open-source framework for actor-oriented software design created over the last 20 years by the Center for Hybrid and Embedded Software Systems (CHESS) of the Electrical Engineering and Computer Science department at the University of California at Berkeley. The aim of the project was to create an environment for the interaction of cyber-physical systems. There have been many contributors, including department faculty, graduate students, and visiting scholars. Ptolemy-II is written in the Java language and communicates with each actor using network sockets.

Each model (actor) is made untestable to the framework via a descriptive XML file (MoML syntax). Each actor could interact directly with other actors or could be made opaque to the environment and managed by a director. The director would appear as an atomic actor to the framework. This approach allows for a hierarchical infrastructure that limits communication among the different actors. A graphical representation of this structure is provided in Figure 8.

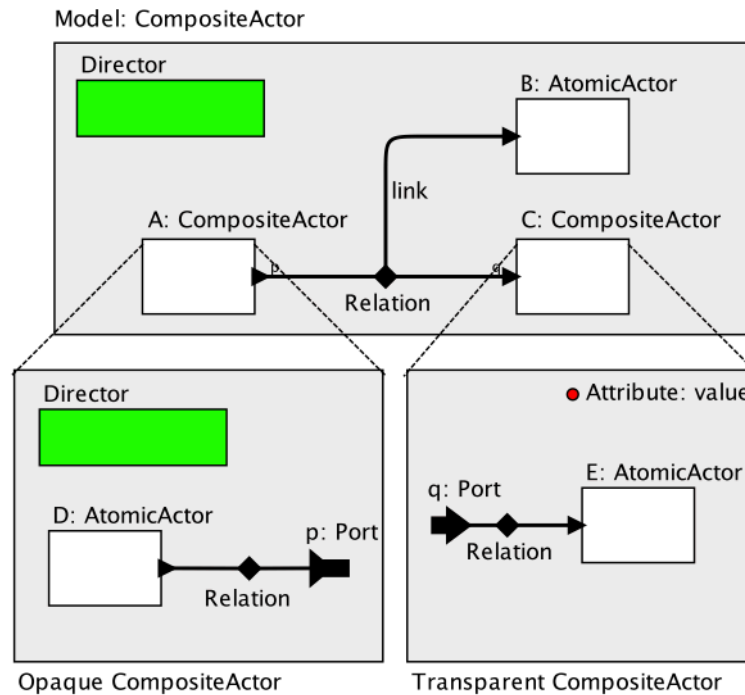


Figure 8. Graphical representation of the hierarchical structure underneath a composite actor.

Ptolemy, being built for time-evolving analysis, also provides a complex model to march the simulation forward in time. Ptolemy allow for no uniform advancement in time of each level of the hierarchical structure and also subcycling. This flexible structure allows the coexistence of actors describing physical phenomena with very heterogeneous time scales without paying the computational price of forcing the whole system to march at the fastest time scale.

4.2.1 Building Controls Virtual Test Bed

Building Controls Virtual Test Bed (BCVTB)²⁶ is both a specialization of Ptolemy-II and a graphical implementation of the framework. It possesses existing interfaces to couple models (actors) implemented in other frameworks/languages. Already coupled software tools include (summarized from the official site <http://simulationresearch.lbl.gov/bcvtb>):

- The EnergyPlus whole building energy simulation program
- The Modelica modeling and simulation environment Dymola
- FMUs for co-simulation and model-exchange for the FIMs 1.0 and 2.0
- The MATLAB and Simulink tools for scientific computing
- The Radiance ray-tracing software for lighting analysis
- The ESP-r integrated building energy modeling program
- The TRNSYS system simulation program
- The BACnet stack, which allows exchanging data with BACnet-compliant Building Automation System (BAS)
- The analog/digital interface USB-1208LS from Measurement Computing Corporation that can be connected to a USB port.

BCVTB also has a graphical user interface that could be used to assemble and edit the component of the simulation environment.

4.2.2 Overview

The following list is an overview of the Ptolemy-II code summarizing its advantages and disadvantages.

Advantages

- Open source
- Already supports most platforms of interest, either directly or through the Modelica FMI
- Source code is available
- Extensible, with many examples to reference
- Created in the DOE laboratory complex
- Has been under continuous development for about 20 years.

Disadvantages

- Significant work remains to be done both in creating interfaces and setting up simulations
- Largely written in Java, which requires a run-time environment
- Potential accuracy issues exist with variable step sizes.

4.3 RAVEN

The RAVEN⁴ project was started at the beginning of 2012 to provide a modern framework for risk evaluation for nuclear power plants. RAVEN possesses extensive sampling capabilities to explore probabilistic behavior of complex systems. This task is facilitated by its generic code interfaces that allow RAVEN to be used as a wrapper for almost any third-party codes. Once RAVEN is wrapped around a generic code, it takes care to run in parallel several instances of the underlying code to speed up the probabilistic analysis. Since it has been designed to be compatible with the most modern codes, RAVEN allows for three levels of parallelisms:

- Batches: each instance of the underlying code is run in parallel
- Message Passing Interface (MPI): each separate instance of the underlying code is allowed to create a parallel environment using the MPI protocol.
- Within each MPI branch the code is allowed to use multi-threaded based parallelism.

This large flexibility in the choice of the parallel environment allows RAVEN to take advantage of large clusters. Some probabilistic (Monte Carlo) and reliability (limit surface) analyses have been performed using RAVEN and RELAP5-3D with more than 3,000 parallel calculations.

RAVEN performs reliability analysis by searching the combinations of system control parameters that satisfy a given constraint on the system behavior. In the case of interest, for the economic assessment of hybrid systems, the limit surface (surface delimiting the values of the parameters where the transition between success/failure of the constraint happens) could be used to identify the capacity and control parameter values that ensure grid reliability. Minimizing the ECE should be performed within the region of space identified by the limit surface. A simple two-dimensional (2-D) example is reported in Figure 9. The example is completely general and demonstrative; Variables 1 and 2 could be seen as two control variables that determine the failure or success of the system with respect to a user-defined constraint. In this case, the success will be determined by passing the reliability grid test; therefore, the success region is the area where seeking for the minimum of the ECE is meaningful (i.e. the area where the grid is capable of passing the reliability test). The blue dots are points that were tested by RAVEN to determine the location of the limit surface (red dots).

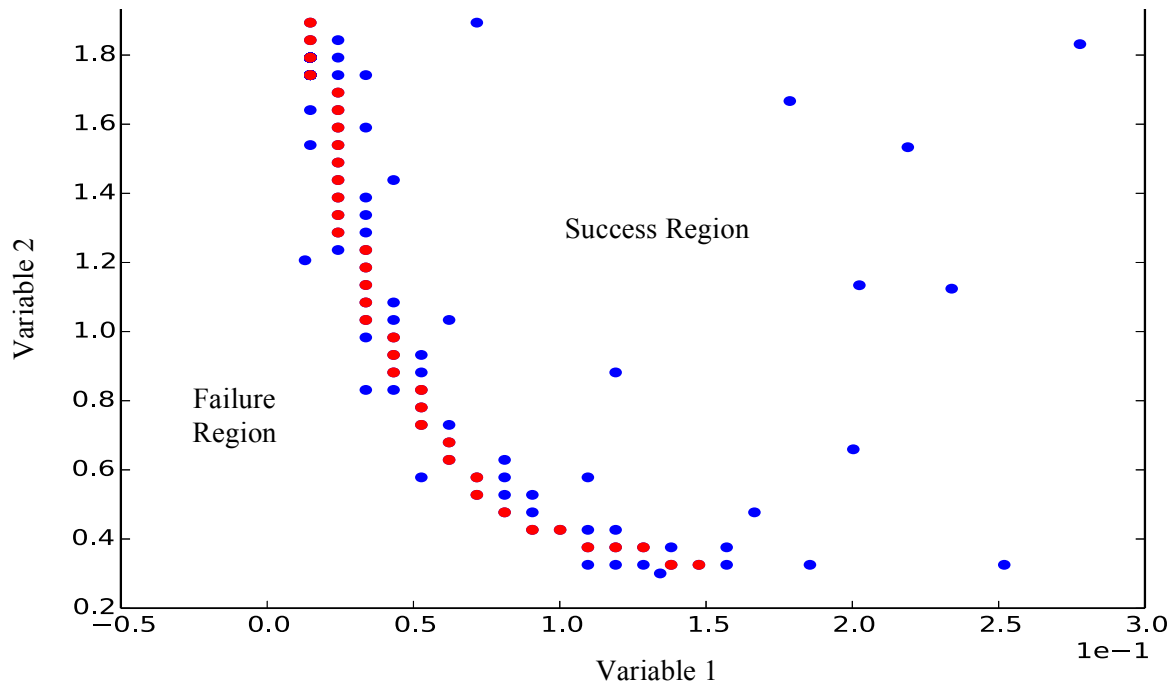


Figure 9. Limit surface representation in a two-dimensional parametric space.

RAVEN uses artificial intelligence based on supervised learning (surrogate models) to accelerate the limit surface search. In particular, all the scikit-learn owned algorithms are implemented along with some algorithms specifically developed for the project. Optimization algorithms are not currently available, but the software infrastructure is already present.

Capability as statistical post-processing are also present in RAVEN, as is the capability to use advanced topological analysis to determine sensitivity. Sensitivity coefficients are the basis for the sensitivity analysis identified as a necessary step in performing a resilience assessment—one of the FOMs discussed in Section **Error! Reference source not found.**

4.3.1 Overview

RAVEN, which is developed in Python and C++, is continuously developed and tested on Linux and MacOS and has also been ported to Windows OS. Currently, it is actively developed; its main development directions are in faster data mining and more scalable algorithms for determining limit surfaces and developing time-dependent surrogate models. While RAVEN possesses generic code interfaces, dedicated interfaces exist for RELAP-7, any MOOSE-based application, and RELAP5-3D. The RAVEN team is currently testing the interface with Modelica.

4.4 Modelica

Modelica is a high-level declarative language for describing mathematical behavior of engineering components that can then be combined into subsystems, systems, and architectures in multiple engineering domains, including electrical, mechanical, chemical, and nuclear. The Modelica modeling (as opposed to programming) language offers numerous benefits for the dynamic modeling, simulation, analysis, and optimization of hybrid energy systems, including:

- Acausal. Capable of solving problems of any structure, both causal (often used for control system design) and acausal (often used in creating schematic oriented physical designs)
 - No *a priori* need to identify knowns and unknowns

- Formulation independent of actual boundary conditions
- Context-independent form, without emphasizing actual solution algorithms while using complex algorithms behind the scenes
- Facilitate model reusability
- Multi-domain. Effective integration of simulated models and physical systems from diverse disciplines
 - Complete integrated simulation including thermo-hydraulics, electrical, mechanical, nuclear, and chemical dynamics of diverse energy conversion systems
 - Support HiL demonstrations
- Open. Allow construction and/or modification of existing component modules to accommodate specific needs, with the language specification being freely available
- Dynamic and Hybrid. Emphasis on dynamic analysis to evaluate and accommodate issues related to flexible operation and variable generation and consumption
 - Dynamic performance and cost analysis, monitoring and controls, sensitivity, robustness, what-if analysis, optimization
 - Time-driven plus event-driven modeling, supporting a wide range of modeling formalisms for continuous and discrete behavior formulated as hybrid differential-algebraic equations
- Non-proprietary. Ease of collaboration through open licensing, without being compelled to pay royalty.

To illustrate its usage in the analysis of hybrid energy systems (HES), Figure 10 shows the top-level of a dynamic model developed in Modelica of a nuclear HES.

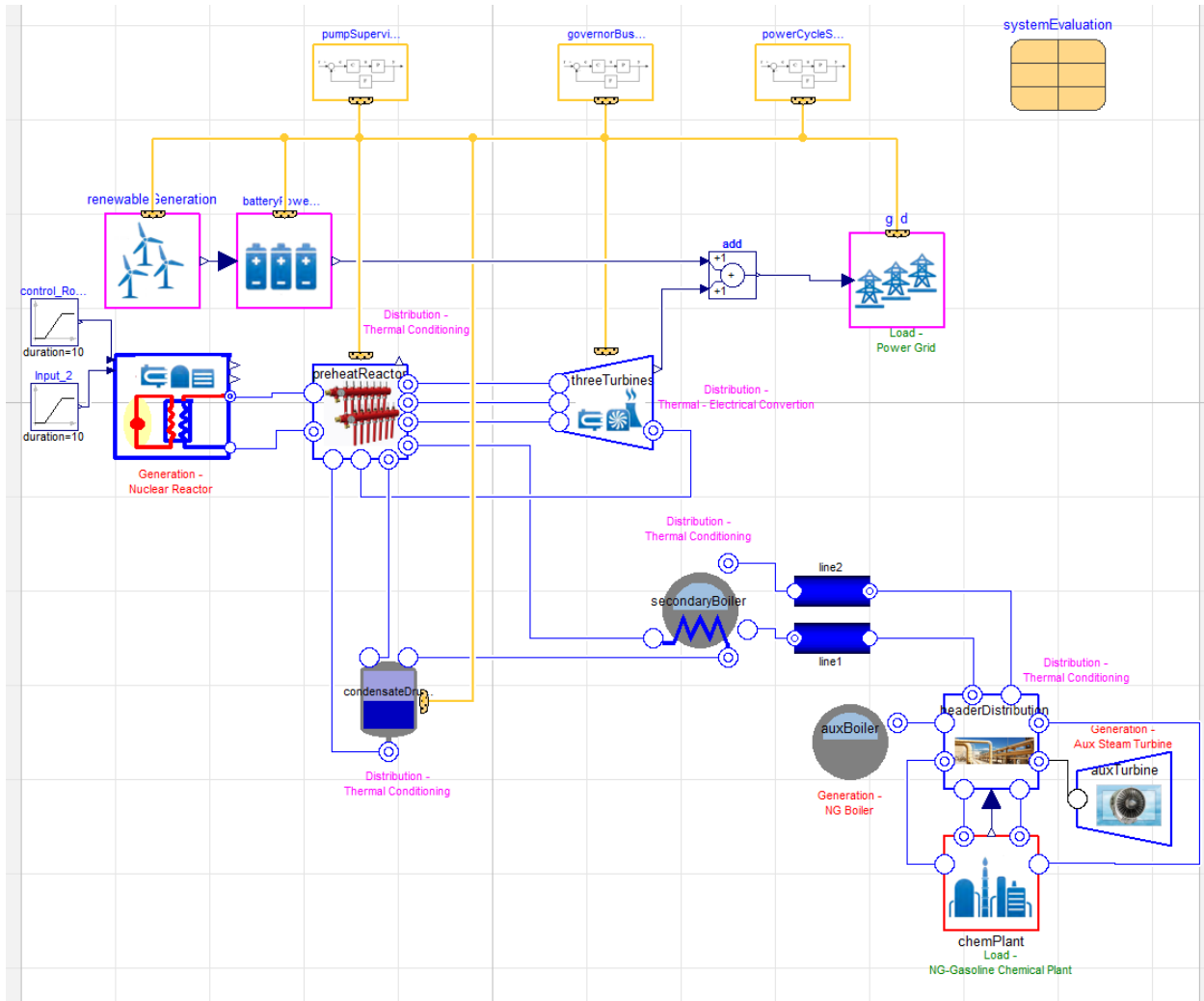


Figure 10: Top-level model for a nuclear hybrid energy system in Modelica.

4.5 RTDS and PSCAD

The RTDS is a real-time ElectroMagnetic Transient (EMT) simulation platform that has the capabilities of performing electric power system simulations with a typical time step of 50 μ s. RTDS uses custom-designed Field-Programmable Gate Array cards as hardware for running the actual mathematical calculations for the simulations. In addition to the real-time simulation capabilities, RTDS also supports HiL simulations. Thus, an actual hardware device such as power electronic inverters, relays, or controller hardware can be interfaced with the RTDS and a controller, or power hardware can be tested as controller HiL (CHiL) or power HiL (PHiL). This provides the ability to test an actual hardware prototype and provides fidelity against modeling errors where a highly detailed model is required for system representation. RTDS supports most communication protocols, such as IEC61850 and DNP3, which can be used for interface communication of HiL in the testing and simulation. RSCAD® is the graphical user interface for modeling and contains an in-built library of power system and control system components. User-defined components can also be designed and used in the modeling using RSCAD. RSCAD uses the run time environment for custom display of simulation results, manual controls, and scripting features for batch files. RTDS can also be used to simulate highly detailed, fast switching power electronic devices with a smaller time steps of up to 2 μ s. This type of simulation can be run in non-real time, but an interface with larger time step simulations is possible for capturing the actual dynamic and transient

response of such models in a detailed EMT simulation. RTDS and RSCAD also offer the feature of being interfaced with other modeling and simulation platforms such as the MATLAB, Simulink, and other custom client programs in C# or other languages. These interfaces have been proven and used for interfacing detailed dynamic thermal models developed on other platforms for electrical-mechanical-thermal co-simulations. One such application has been demonstrated at INL in Panwar et al.'s 2015 conference proceeding²⁷ for the co-simulation of HES in an area power system for dynamic and transient analysis using RTDS and MATLAB. The simulation setup is shown in Figure 11 below.

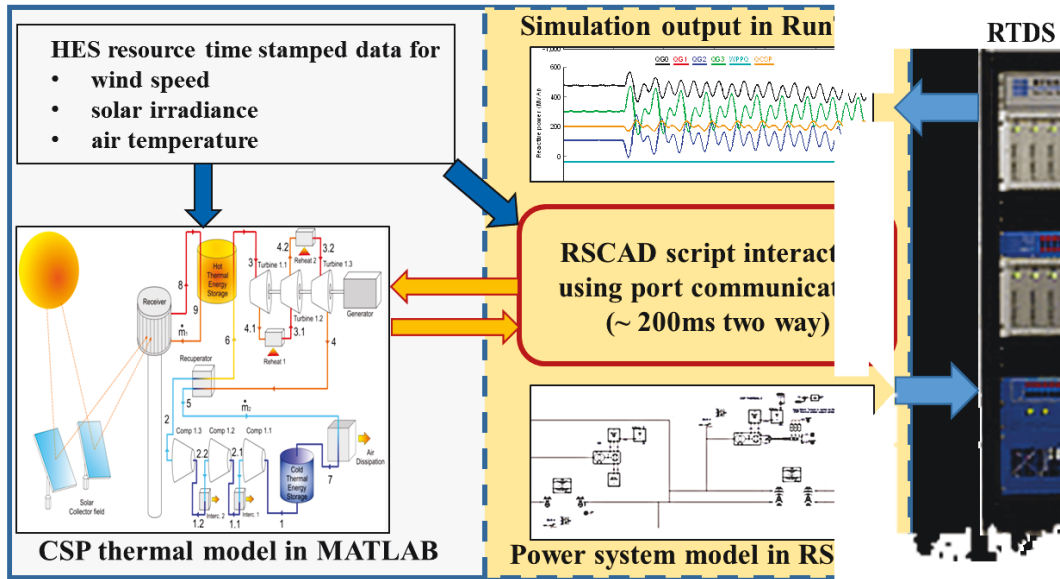


Figure 11. Simulation setup with RTDS and integrated HES model.

RTDS racks at the INL Energy Systems Laboratory are also being used to perform at-scale battery, vehicle charging station, and grid integration testing with detailed EMT models. This provides an opportunity for large-scale simulations including various HiL and models on different platforms interfaced with RTDS. The detailed conceptual setup is shown in Figure 12.

4.6.1 General Code Simulation Capabilities

RELAP5-3D has multi-dimensional thermal-hydraulic, neutron kinetic, and heat transfer modeling capabilities, as well as models for trips and controls. With these features, users have constructed input models for a wide variety of designs and devices, including nuclear reactors, experimental facilities, steam delivery systems, jet engines, and other applications. However, the primary purpose has been nuclear industry applications. Its extensive use in nuclear power plant design, operational and accident scenarios, and operator training simulators by industry, research institutions, and universities has produced a fast-running, reliable, and robust computer program that is used more than any other code for these purposes.

Since nuclear plants have large sections of piping with essentially one-dimensional (1-D) flow, RELAP5-3D provides modelers with many 1-D components to apply where engineering judgment indicates they will provide sufficient fidelity. Use of these component models (pumps, valves, separators, branches, etc.) produces fast runtimes. The multi-dimensional component in RELAP5-3D allows the user to accurately model the multi-dimensional flow behavior that can be exhibited in any component or region of a nuclear reactor coolant system. There is also 2-D conductive and radiative heat transfer capability and modeling of plant trips and control systems. RELAP5-3D allows for the simulation of the full range of reactor transients and postulated accidents, including:

- Operational transients
- Startup and shutdown
- Maneuvers (e.g., change in power level, starting/tripping pump)
- Small and large break Loss Of Coolant Accidents (LOCA)
- Anticipated Transient Without Scram (ATWS)
- Loss of offsite power
- Loss of feedwater
- Loss of flow.

In addition, RELAP5-3D has been used to model virtually every major type of reactor:

- Light Water Reactors including:
 - Pressurized water reactor (PWR), advanced PWR (APWR)
 - Boiling water reactor (BWR), advanced BWR (ABWR), etc.
- Heavy water reactors (e.g., Canada Deuterium Uranium [CANDU] reactor)
- Gas-cooled reactors:
 - Very high temperature reactor (VHTR),
- Liquid metal cooled reactors
- Molten-salt cooled reactors.

To develop RELAP5-3D plant models that run these transients, many practical tools and code features are available to aid the user. First is the SNAP³¹ (Symbolic Nuclear Analysis Package) Graphical User Interface (GUI), available to help users create input models. Users create a 2-D representation of the plant model by dragging and dropping icons of plant components onto a canvas. Default information for each component can be overridden via value entry widgets accessed through a menu system.

For both operational and accident scenarios, numerous RELAP5-3D plant models already exist and may serve as examples, or a starting point, for developing new plant models. Models exist for various kinds of PWRs, BWRs, gas-cooled reactors, liquid metal reactors, and molten salt reactors. Many of these are proprietary to the company that developed the input model; however, generally one or two are available for any type of plant. For small modular reactors (SMR), the models available are proprietary to the particular SMR company.

4.6.2 Computational Runtime

The runtime for a plant model depends on the size of the model, the length of the transient, and the time step size. In RELAP5-3D, advancement in time may fail for various reasons such as excess mass error, Courant limit violation, excessive increase in pressure in a computational cell, and physical property out of range. An adjustment is made, generally the halving of the time step size, and the advancement is repeated. RELAP5-3D input models made for operator training simulators run faster than real time. These models of real plants provide sufficient fidelity for operators in the simulator to experience similar action to what the actual reactor exhibits. Plant models used for design and analysis of power plants may require greater fidelity and may run slower than real time.

While the time resolution of RELAP5-3D most of the time is probably above the range of 10^{-5} sec, it nonetheless captures all physical time scale characteristics of the nuclear reactor system, which is a natural candidate for the grid stability analysis.

4.6.3 Code Pedigree

To continue at the forefront of nuclear power plant modeling, RELAP5-3D verification and validation has continued to evolve with the code requirements and industry needs.

A form of validation testing, developmental assessment (DA),³² is applied to validate RELAP5-3D performance for generic nuclear safety analyses. DA checks RELAP5-3D calculations against analytical solutions and compares data from separate-effects and integral tests against code calculations. Each new RELAP5-3D code release is reevaluated with DA and published as Volume 3 of the RELAP5-3D manuals. The verification test suite contains 61 input models of analytical problems and special test cases that cover more than 200 code features identified as important to nuclear power plant applications. A dozen of these input models verify the coupling ability of RELAP5-3D through the PVM Executive.

4.7 Aspen

Aspen Plus and Aspen HYSYS are commercial process modeling software by Aspen Technology, Inc. Both software packages are used to develop complex chemical and thermos-fluid systems by providing unit operations that simulate components such as pumps, heat exchangers, chemical reactors, pipes, and chemical columns (distillation, adsorption, etc.). Large databases of thermodynamic and chemical properties are available with mixing models, which allows the software to simulate accurate multiple component and phase gases, fluids, and solids such as oil and coal. Processes such as natural gas or coal to liquid fuels, ammonia, power conversion cycles, methanol production, and high temperature electrolysis have been developed using the software. The software is widely used in oil and gas and chemical industries.

The Aspen software runs on a Microsoft Windows® platform. The Aspen Simulation Workbook allows the software to actively interact with Microsoft Excel. However other means of interacting with other software is not apparent. User customized unit operations may be either programmed using Fortran, the Aspen Simulation Workbook or in the case of HYSYS, an embedded HYSYS spreadsheet. An example of this is the unit operation for the electrolysis process. For HYSYS, the electrolysis process is simulated by the embedded spreadsheet and as a FORTRAN subroutine within Aspen Plus. Aspen uses flow sheets to breakdown complex operations into simpler subsystems that interact with each other. Figure 13 is an example of such a system. The flow sheet to the left is the support equipment for high temperature electrolysis. The flow sheet to the bottom right is a subflow sheet that simulates the splitting of the water and includes the power and heat needed to do so.

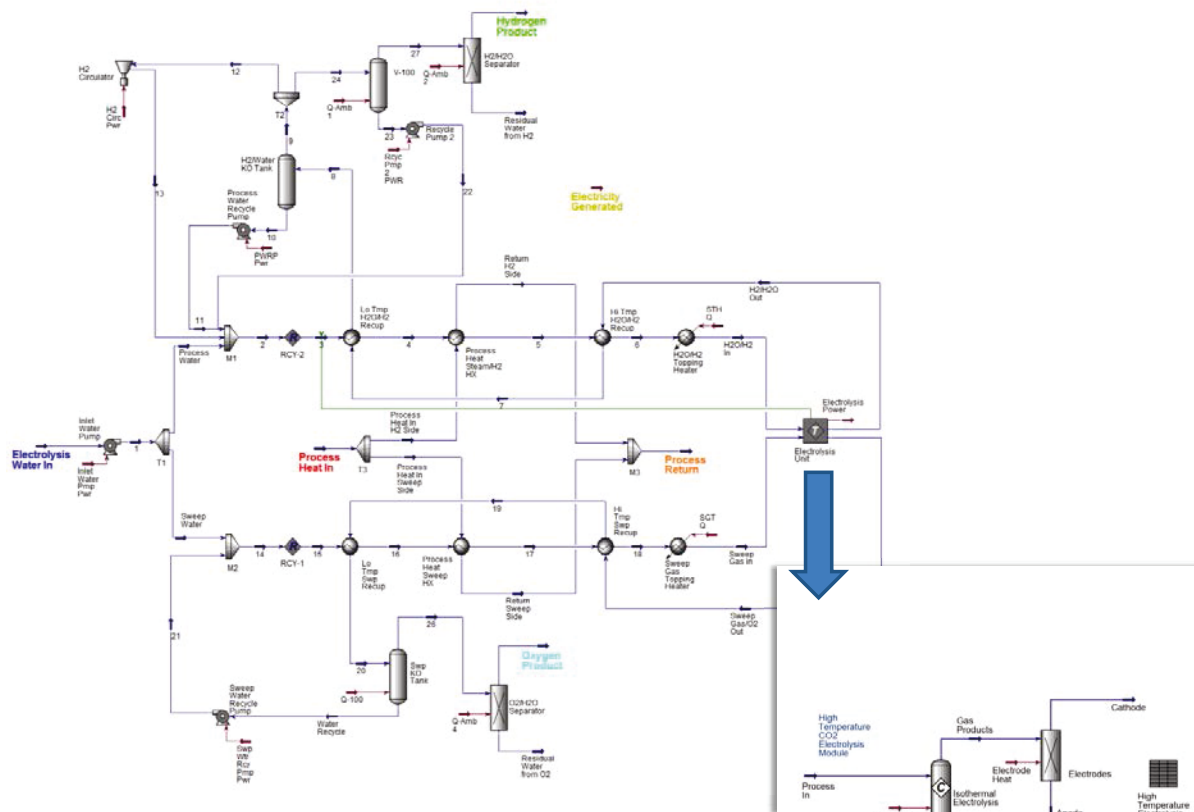


Figure 13. High temperature steam electrolysis support flow sheet with electrolysis subflow sheet.

The software is of both steady state and transient modeling. Process control functions are also available to develop and simulate control strategies. Aspen provides an interface that allows the software to directly monitor and control actual processes.

Aspen provides a suite of software that includes economic tools to estimate the process capital and operation and maintenance costs, detailed heat exchanger design, electrolytic solution simulation, complex piping, and equipment sizing and design.

5. SOFTWARE DEVELOPMENT PROCESS

As already outlined, this is a complex and large effort that will need to be carried out across laboratories, and several teams will take part in the development. Moreover, the final product will use several different computational platforms, hardware, and languages. For this reason, it is necessary to enforce from the beginning a strong and common development approach.

5.1 Software Lifecycle Management and Quality Assurance

The software lifecycle management, in modern approaches, is strictly connected with the level of Quality Assurance (QA) requested for the final product. Many of the steps required for QA purposes could be incorporated in the software development process, making the QA process less expensive and the development process more robust. Experience shows that if a project's complexity increases, deployment time of new features and maintenance costs also increase more than linearly. First, it is necessary to identify the QA target level. Several QA standards and levels have been defined over time and in different fields. INL is bounded, by its DOE contract, to operate under the ASME NQA-1 prescription. Within the ASME NQA-1, different levels of rigor exist depending on the impact a malfunction would have. Moreover, each specific DOE-funded program might have specific (more or less strict) QA requirements that supersede the general contract between DOE and INL. This approach ensures that QA requirements are common across projects within the same program, even if tasks are executed by different organizations. In this case the situation might be even more fragmented because projects like RAVEN are supported by different programs. To be able to leverage products generated in different programs, it is necessary to select the strictest standards for all programs involved. The work proposed here is under the Advanced Reactor Technologies QA document.³³

It is foreseeable that for the initial part of the project, the outcome will be classified as a research product and thus it will be subject to less stringent requirements. In INL graded approach implementation of NQA-1, this stage corresponds to a Quality Level Determination (QLD) of three. For convenience we will use a graded approach as defined in the INL document³⁴ since the Advanced Reactor Technologies (ART) QA documents hand off to this document; other DOE programs have a very similar grading system (e.g., LLNL-SM-455533³⁵).

1. The software is accessible to all developers in a remote repository like GitHub (<https://github.com/>) if open source, or GitLab (<https://gitlab.com/>) otherwise.
2. Each developer will start a development phase by downloading the latest version of the software and create locally, in his machine, the sought development.
3. When that development is finished, the developer will submit a “merge request” to another member of the team who will act as reviewer and approve or disapprove the merge request.
4. If software modifications are accepted, an automatic system will merge the modifications with the latest version of code available in the repository and test for incompatibility (regression testing).
5. If no compatibility issues are detected, the modifications are merged in the main repository and made available to the whole development team.
6. The cycle starts back from Point 1.

To clarify, the merge request should include three components:

1. The software containing the modification
2. The input and outputs of simulations that specifically test the new development
3. The corresponding change to the software manual.

The review by a second developer tests compliance with the following requirements:

- All three above-mentioned components should be present
- The code syntax should conform to the code standards
- The inputs provided extensive testing of the new part of the software using automatic tools. The common practice is to accept between 80% to 85% as minimum coverage.

In addition to the above procedures and documentation, a bug tracking system is needed. Each user has access to a ticketing system where the bugs should be described and assigned to the development team. The development team has ~20% of its weekly time allocated to address those issues.

The above process is actually not sufficient to meet the requirements for QLD 2, which is a reference level for any non-safety related software (safety related applications require QLD 1), and should be the long-term target level for the software developed under this project. In addition, a software requirement document should be maintained. Before starting any new task, each developer should review the document, describing in detail which features the new development will bring, how the scope will be achieved (algorithm description), and detailing the action of each new software component (corresponding software infrastructure). We suggest adding atomistic tests to the regression test set. Those tests focus on single software components separately.

While this process might appear overwhelming, it is the experience of the author as principal investigator and team member of several large-scale software development projects that the more these principles are enforced, the more resilient the project is in the long term. It avoids the exponential growth of the maintenance time with respect to the growth of the software capabilities, helps to maintain the code function across different operative systems and their updates, allows a better scalability of the development team (the completion time of the project is more close to be inversely proportional to the number of team members), and at the end provides a clear picture of the software development to the project manager. While the code validation is not a QA issue per se (QA ensures the traceability and accountability, not the accuracy), the infrastructure described could be used to deploy an automatic validation system. In this respect BISON (the fuel-performance code based on MOOSE) is pioneering this approach. Every time a new code commitment is completed, regression tests are performed, and also once a week a set of validation cases can be run for a new validation assessment.

Constructing software infrastructure to manage the above-described process is an expensive and time-consuming task. Currently, all the MOOSE-based applications benefit from an environment capable of performing the necessary tasks; therefore, the authors suggest cloning or somehow leveraging such an environment.

6. CONCLUSION

The FOMs identified as relevant for the determination of the value of the introduction of hybrid systems in the energy supply mix¹ are several and heterogeneous. The current work to define a strategy for modeling, simulation, and control of candidate hybrid energy systems takes into account the electricity cost, the grid reliability, and the resilience of the supply mix. Usually those figures of merit are analyzed with a certain degree of separation that makes their evaluation more feasible but, at the same time, this choice might lead to an optimization process that is less effective than what could be achieved by considering the whole set of FOMs at once. Here the optimization process refers to the search of the optimal configuration of the system with respect the electricity supply mix, production of other commodities (e.g. hydrogen), and operational control parameters of the individual components or subsystems within the system.

This report proposes a strategy for analyzing the system economic feasibility accounting for grid reliability requirements and performing a posterior evaluation of the resilience. The computational challenges that arise from such an approach are several and span from uncharted territory in optimization under uncertainty of large dynamic systems to the capabilities of properly simulating the various components within the systems. The major algorithm needs are identified and a short review of their current status of knowledge is provided. The report also suggests that, given the large span of time scales at which the system needs to be analyzed, it could be possible to separate the time scales to achieve a reduction in computational costs. This would require using different models of the system components depending on the time scale to be simulated. The different software, modeling platforms, and hardware that are currently considered for building the numerical models of the system components and the optimization process are reviewed in the attempt to highlight their advantages and disadvantages in the considered context. In the last part of this report the difficulties of accomplishing such a large and complex simulation are projected into the needs of a proper software development environment that should allow coordination among several teams, and to meet the proper level of QA in the final product.

7. REFERENCES

- ¹ S. Bragg-Sitton, R. Boardman, M. Ruth, O Zinaman, C. Forsberg, J. Collins, *Integrated Nuclear- Renewable Energy Systems: Foundational Workshop Report*, INL/EXT-14-32857 Rev. 1, NREL/TP-6A20-62778, August 2014.
- ² The RELAP5-3D© Code Development Team, *RELAP5-3D© Code Manual Volume I: Code Structure, System Models and Solution Methods*, INEEL-EXT-98-00834, Revision 2.4, June 2005.
- ³ R. A. Berry, J. W. Peterson, H. Zhang, R. C. Martineau, H. Zhao, L. Zou, D. Andrs, *RELAP-7 Theory Manual*, INL/EXT-14-31366, Revision 1, March 2015.
- ⁴ C. Rabiti, A. Alfonsi, J. Cogliati, D. Mandelli, R. Kinoshita, S. Sen, *RAVEN User Manual*, INL/EXT-15-34123, March 2015.
- ⁵ D. Gaston, G. Hansen, S. Kadioglu, D. A. Knoll, C. Newman, H. Park, C. Permann and W. Taitano “Parallel multiphysics algorithms and software for computational nuclear engineering,” *Journal of Physics: Conference Series*, Vol.180, 2009.
- ⁶ MATLAB, *The Language of Technical Computing*, <http://www.mathworks.com/help/matlab/index.html>.
- ⁷ SIMULINK, *Simulation and Model-based Design*, <http://www.mathworks.com/products/simulink/>.
- ⁸ H. Elmqvist, S. E. Mattsson “MODELICA — “The Next Generation Modeling Language An International Design Effort,” *Proceedings of the 1st World Congress on System Simulation (WCSS’97), September 1–3, 1997, Singapore*.
- ⁹ Aspen Technology, Inc., “Aspen Plus®, Aspen Plus User Guide, Version 10.2,” February 2000.
- ¹⁰ Claudius Ptolemaeus, Editor, *System Design, Modeling, and Simulation using Ptolemy II*, org., 2014, <http://ptolemy.org/systems>.
- ¹¹ Manitoba HVDC Research Centre, *USER’S GUIDE on the use of PSCAD*, Power Systems Computer-Aided Design, 2010.
- ¹² F. E. Cellier, H. Elmqvist, “Automated Formula Manipulation Supports Object-Oriented Continuous-System Modeling,” *IEEE Control Systems*, pp. 0272–1708, 1993.
- ¹³ E. Ibanez and M. Milligan, *Comparing Resource Adequacy Metrics*, NREL/CP-5D00-62847, September 2014.
- ¹⁴ E. Ibanez and M. Milligan, *A Probabilistic Approach to Quantifying the Contribution of Variable Generation and Transmission to System Reliability*, NREL/CP-5500-56219, September 2012.
- ¹⁵ U.S. Energy Information Administration (EIA), *Levelized Cost of Electricity and Levelized Avoided Cost of Electricity Methodology Supplement*, July 2013.
- ¹⁶ B. J. Kirby, *Frequency Regulation Basics and Trends*, ORNL/TM-2004/291, December 2004.
- ¹⁷ Wes McKinney and PyData Development Team, *Pandas: Powerful Python Data Analysis Toolkit*, March 2015.
- ¹⁸ S. K. Aggarwal, L. M. Saini, A. Kumar, “Electricity price forecasting in deregulated markets: A review and evaluation,” *Electrical Power and Energy Systems* 31, 2009, pp.13–22.

-
- ¹⁹ N. V. Queipoa, R. T. Haftkaa, W. Shyya, T. Goela, R. Vaidyanathana, P. K. Tuckerb, “Surrogate-based analysis and optimization,” *Progress in Aerospace Sciences*, Vol. 41, 2005, pp. 1–28.
- ²⁰ P. Denholm, R. Margolis, B. Palmintier, C. Barrows, E. Ibanez, L. B. J. Zuboy, *Methods for Analyzing the Benefits and Costs of Distributed Photovoltaic Generation to the U.S. Electric Utility System*, NREL/TP-6A20-62447, September 2014.
- ²¹ R. Sioshansi, P. Denholm, *The Value of Concentrating Solar Power and Thermal Energy Storage*, NREL-TP-6A2-45833, February 2010.
- ²² H. E. Garcia, J. Chen, J. S. Kim, M. G. McKellar, W. R. Deason, R. B. Vilim, S. M. Bragg-Sitton, R. D. Boardman, *Nuclear Hybrid Energy Systems - Regional Studies: West Texas & Northeastern Arizona*, INL/EXT-15-34503, April 2015.
- ²³ T. Blochwitz, M. Otter, M. Arnold, C. Bausch, C. Clauß, H. Elmqvist, A. Junghanns, J. Mauss, M. Monteiro, T. Neidhold, D. Neumerkel, H. Olsson, J.-V. Peetz, S. Wolf, “The Functional Mockup Interface for Tool Independent Exchange of Simulation Models,” *Proceedings 8th Modelica Conference, Dresden, Germany, March 2011*.
- ²⁴ J. Eker, J. Janneck, E. A. Lee, J. Liu, X. Liu, J. Ludvig, S. Sachs, Y. Xiong, “Taming heterogeneity – the Ptolemy approach,” *Proceedings of the IEEE*, 91(1): pp. 127–144, January 2003.
- ²⁵ T. Dahlgren, D. Ebner, T. Epperly, G. Kumfert, J. Leek, A. Prantl, *Babel Users’ Guide*, LLNL UCRL-SM-230026, January 2012.
- ²⁶ M. Wetter, “Co-simulation of building energy and control systems with the Building Controls Virtual Test Bed,” *Journal of Building Performance Simulation*, Vol. 4, No. 3, 2011, pp. 185–203.
- ²⁷ M. Panwar, M. Mohanpurkar, J. D. Osorio, R. Hovsapien, “Significance of Dynamic and Transient Analysis in the Design and Operation of Hybrid Energy Systems,” *Proceedings of the 9th International Topical Meeting on Nuclear Plant Instrumentation, Control, and Human Machine Interface Technologies (NPIC&HMIT 2015)*, February 2015, Charlotte, North Carolina, USA.
- ²⁸ W. L. Weaver, E. T. Tomlinson, D. L. Aumiller, “A PVM Executive Program for Use with RELAP5-3D,” *Proceedings of Conference*, pp. 1057-1061, ICONE10-22678, Arlington, Virginia, April 14-18, 2002, American Society of Mechanical Engineers.
- ²⁹ C. Rabiti, “PHISICS Reactor Physics Package in RELAP5-3D,” *RELAP5-3D Newsletter 2nd Quarter 2012*, Idaho National Laboratory, http://www.inl.gov/relap5/news/2012_2q_r5_3d_physics_in_r5.pdf.
- ³⁰ B. L. Smith, “Assessment of CFD Codes Used in Nuclear Reactor Safety Simulations,” *Nuclear Engineering and Technology*, Vol. 42, No. 4, August 2010.
- ³¹ K. R. Jones, *Symbolic Nuclear Analysis Package (SNAP) User’s Manual, Version 2.2.1*, Applied Programming Technology, Inc., Bloomsburg, Pennsylvania, 17815-1951, October 25, 2012.
- ³² The RELAP5-3D Code Development Team, *RELAP5-3D Code Manual Volume III: Developmental Assessment*, INEEL/EXT-98-00834, Revision 4.2, June 2014.
- ³³ PLN-2690, “VHTR Technology Development Office Quality Assurance Program Plan,” Idaho National Laboratory, July 28, 2011.
- ³⁴ LWP-13621, “Software Quality Assurance for Research and Development Activities,” Idaho National Laboratory, Rev. 1, March, 2013.

³⁵ LLNL-SM-455533, “NEAMS Software Quality Assurance Program,” Los Alamos National Laboratory, April, 2013.

8. APPENDIX

Appendix A, Summary of Modeling Tools

Appendix A

Summary of Modeling Tools

Name	Type	License	Description	Language	Operative System
RELAP5-3D	Physical model	Commercial (DOE free)	Software capable to perform transient simulation of Nuclear power plants	Fortran 90	Windows/Linux
RELAP-7	Physical model	DOE free	Software capable to perform transient simulation of Nuclear power plants	C++	Linux/MacOS
RAVEN	Optimization and probabilistic analysis framework	DOE free	Perform statistical analysis and optimization of physical models represented by codes	Python/C++	Linux/MacOS/Windows
MATLAB/Simulink	Development environment	Commercial	Development environment with a large set of libraries and interfaces for other languages and also external hardware	N/A	Linux/MacOS/Windows
Modelica	Development environment	Free	Language	N/A	Linux/MacOS/Windows
Dymola	Development environment	Commercial	Compiler for Modelica	N/A	Linux/MacOS/Windows
Aspen	Simulation environment	Commercial	Simulation environment for chemical process optimization software	N/A	
Ptolemy	Software platform	Free	Computational actor based framework	Java	Linux/MacOS/Windows
PSCAD	Software platform	Commercial	Graphical tool to develop program (models) for RTDS and for interfacing RTDS with external simulation environments	N/A	Windows
RTDS	Hardware platform	N/A	Hardware platform with very low OS support designed for reproducing physical devices by low level programming	N/A	N/A