

# Light Water Reactor Sustainability Program

## Development and Application of a Risk Analysis Toolkit for Plant Resources Optimization



September 2020

U.S. Department of Energy

Office of Nuclear Energy

**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, do 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.

# **Development and Application of a Risk Analysis Toolkit for Plant Resources Optimization**

**D. Mandelli, C. Wang, M. Abdo, A. Alfonsi, P. Talbot, J. Cogliati, C. Smith (INL)  
D. Morton (Northwestern University)  
I. Popova (Texas State University)  
S. Hess (Jensen Hughes)**

**September 2020**

**Prepared for the  
U.S. Department of Energy  
Office of Nuclear Energy**

## EXECUTIVE SUMMARY

This report summarizes the R&D activities of the Risk-Informed Asset Management (RIAM) project during fiscal year 2020 (FY-20). This project focuses on the development of computational methods designed to optimize plant operations (e.g., maintenance and replacement schedule, optimal maintenance posture) provided system and component health and cost data. This project development lives in cooperation with the Plant Health Management (PHM) project which focuses on the development of methods that integrate component health data and propagate such information at the system level to evaluate most relevant sources of risk. This year's activities for the RIAM project focused on the continuation of the development of schedule optimization algorithms. While in FY-19 we focused on both deterministic and stochastic capital budgeting methods, in FY-20 we moved forward by implementing two versions of schedule optimization methods. The first one reformulates the capital budgeting problem in a distributionally robust form which allows the user to rely on data directly rather than proposing a distribution from the data itself. The second version reformulates the capital budgeting explicitly using risk measures as variables to maximize or minimize. Lastly, we focused on the development of methods designed to identify the optimal maintenance posture based on the Pareto frontier analysis. Rather than performing a tradeoff analysis (i.e., identify the absolute best posture), the Pareto frontier analysis performs a trade space exploration approach (i.e., identify value and costs of several postures and have the analyst perform the task of imposing desired value and cost constraints). This is performed by identifying maintenance postures that maximize value (e.g., system availability) and minimize operational costs (i.e., the Pareto frontier in a value-cost trade space). Lastly, we have developed discrete model-based optimization methods based on evolutionary algorithms, i.e., genetic algorithms. This approach allows us to optimize maintenance schedule when system availability and system cost are tightly coupled to each other.

# CONTENTS

EXECUTIVE SUMMARY.....	2
ACRONYMS.....	9
1. INTRODUCTION.....	11
2. COST AND RISK CATEGORIZATION R&DD PATH.....	12
3. METHODS FOR PLANT RESOURCES MANAGEMENT.....	14
4. INVESTMENT EVALUATION.....	15
5. PARETO FRONTIER ANALYSIS.....	18
5.1 Pareto Frontier Analysis for RIAM Applications.....	19
5.1.1 Identification of Plant/System Optimal Maintenance Posture.....	20
5.1.2 Identification of the Optimal Set of Plant/System Maintenance Operations.....	21
5.2 Pareto Frontier Post-Processor.....	22
5.3 Pareto Frontier Test Cases.....	22
5.3.1 Case 1: System Reliability Modeling.....	23
5.3.2 Case 2: Maintenance Reliability Modeling.....	27
5.4 Maintenance Schedule Optimization.....	30
6. MODEL-BASED OPTIMIZATION METHODS.....	33
6.1 Continuous Methods.....	35
6.1.1 Gradient-Based Methods.....	35
6.1.2 Stochastic Methods.....	36
6.2 Discrete Optimization Methods.....	36
6.2.1 Genetic Algorithms.....	37
6.2.2 GA Data Structures.....	38
6.2.3 GA Workflow.....	39
6.3 GA Development.....	41
6.4 RIAM Applications for GA methods.....	41
6.4.1 Knapsack Problem.....	41
6.4.2 Knapsack Problem (Modified).....	42
6.4.3 Multiple-Knapsack Problem.....	43
6.4.4 Multiple-Knapsack Problem (Modified).....	43
6.4.5 Traveling Salesman Problem.....	44
6.5 Initial Testing of GA Methods.....	44
7. METHODS DEVELOPMENT: LOGOS.....	46
8. STOCHASTIC OPTIMIZATION WITH CVAR.....	49

8.1	Deterministic Capital Budgeting.....	49
8.1.1	Stochastic Capital Budgeting with Options .....	51
8.2	Risk-Based Stochastic Capital Budgeting using Conditional Value-at-Risk.....	58
8.2.1	Definitions of VaR and CVaR .....	58
8.2.2	CVaR in Capital Budgeting .....	59
8.2.3	CVaR Mathematical Optimization Model .....	60
8.2.4	Analysis of Risk Versus Return Optimization using CVaR and NPV .....	61
8.2.5	CVaR for Single Knapsack Problem .....	67
8.2.6	CVaR for Multi-Dimensional Knapsack Problem.....	67
8.2.7	CVaR for Multiple Knapsack Problem.....	67
8.2.8	CVaR for Multiple-Choice Knapsack Problem .....	68
9.	DISTRIBUTIONALLY ROBUST OPTIMIZATION .....	68
9.1	Overview .....	68
9.2	Defining a Distributional Uncertainty Set via the Wasserstein Distance .....	69
9.3	Towards a Computationally Tractable Reformulation.....	70
9.4	A Computationally Tractable Reformulation.....	72
9.5	Tractable Reformulation Specialized to Stochastic Capital Budgeting .....	72
9.6	Analysis of Risk versus Return Optimization using Distributionally Robust Optimization.....	73
9.6.1	DRO for Single Knapsack Problem .....	78
9.6.2	DRO for Multi-Dimensional Knapsack Problem.....	79
9.6.3	DRO for Multiple Knapsack Problem .....	79
9.6.4	DRO for Multiple-Choice Knapsack Problem.....	80
10.	LINK WITH PHM PROJECT.....	80
11.	PHM-RIAM WORKFLOW .....	81
12.	LINKING MAINTENANCE APPROACHES WITH OPTIMIZATION METHODS.....	83
13.	CONCLUSIONS .....	84
	REFERENCES.....	86

## FIGURES

Figure 1. High level description of the RI-PSH platform. ....	12
Figure 2. Interactions among the RIAM project and other DOE projects in the development of the RI-PSH platform. ....	13
Figure 3. Structure of the RI-PSH computational platform. ....	15
Figure 4. Classification of the optimization algorithms developed under the RIAM project based on employed data structure and the method being used. ....	16
Figure 5. Reliability vs. NPV plot for evaluation of candidate projects [19]. ....	17
Figure 6. Set of options (blue dots) plotted in a cost vs. utility space. ....	18
Figure 7. Pareto frontier obtained from a set of options plotted in a cost vs. utility space. ....	19
Figure 8. Propagation of uncertainties for the points on Pareto frontier and imposition of cost and utility constraints. ....	19
Figure 9. Pareto frontier analysis in an availability vs. cost space. ....	20
Figure 10. Graphical representation of the considered HPI system. ....	22
Figure 11. Graphical representation of the RAVEN EnsembleModel for a generic system maintenance scheduling optimization problem. ....	24
Figure 12. Pareto frontier obtained for case 1. ....	25
Figure 13. Zoom of the Pareto frontier obtained for case 1. ....	25
Figure 14. Pareto frontier obtained for case 2. ....	28
Figure 15. Graphical representation of the continuous optimization problem. ....	34
Figure 16. Model-based optimization scheme. ....	35
Figure 17. Data structures and methods employed in genetic algorithms that can be employed to develop evolutionary programs. ....	38
Figure 18. Graphical representation of the GA data structures. ....	39
Figure 19. GA workflow. ....	40
Figure 20. Single point crossover operation. ....	40
Figure 21. Two-points crossover operation. ....	41
Figure 22. Example of a single bit mutation of a chromosome. ....	41
Figure 23. Data workflow for the GA methods developed in RAVEN. ....	42
Figure 24. GA workflow specified in RAVEN for the multiple knapsack problem test case. ....	46
Figure 25. Snapshot of the LOGOS repository. ....	47
Figure 26. Candidate projects (e.g., 16 projects) are listed in the left-hand column. In a priority list these are mapped to priority levels in the middle column. Then the uncertain budget is revealed. For the budget realized in the figure, the two projects with the highest priority levels can be implemented. ....	53

Figure 27. Histogram of NPV realizations across 90 scenarios accounting for the probability mass of each scenario. The optimized priority list maximizes the expected net present value but does not account for risk.....	58
Figure 28. Relationship between value-at-risk and conditional value-at-risk. A typical value of $\alpha$ is $\alpha=0.90$ . .....	59
Figure 29. Flow-chart illustrating two principled ways to approach risk-averse optimization. In both cases we start with available data. The left path then fits a probability distribution to that data, and then select a risk-averse objective function. One popular choice is a weighted sum of expectation and conditional value-at-risk. We then minimize risk, after possibly forming a Monte Carlo sample average approximation or employing some other discretization of the probability distribution. The right-hand path instead formulates and solves a distributionally robust optimization model. We defer detailed discussion of the right-hand path to Section 11.....	60
Figure 30. Subset of the Pareto frontier using just $\lambda=0,0.5$ , and 1, where the risk is measured by CVaR and return is measured by Expected NPV. Optimal solutions are obtained by solving model (1) with input from Table 17. The plot is oriented so that risk grows moving left-to-right on the x-axis and return grows moving bottom to top along the y-axis. Starting from the upper-right point, we can reduce risk by \$2.18M by reducing expected NPV by \$1.22M. Additional reduction is risk of \$0.33M requires reducing NPV by an additional \$1.17M. ....	63
Figure 31. Histogram for scenario NPVs accounting for probability mass associated with each scenario. The red histogram corresponds to $\lambda=0.05$ when most weight is on maximizing NPV and the blue histogram corresponds to $\lambda=0.95$ when most weight corresponds to minimizing CVaR for low NPV realizations. ....	64
Figure 32. Pareto frontiers for different values of $\alpha$ . The figure repeats the Pareto frontier for $\alpha=0.75$ (left-most plot) from Figure 30 except that now we include the full range of values of $\lambda$ and achieve five rather than three points on the efficient frontier. The additional four plots repeat similar frontiers but for $\alpha=0.8,0.9,0.95,0.99$ . where we focus on values that are further in the poor-outcome tail of NPV. Note that as the figure shows, there are specific NPV values that repeat across different values of $\alpha$ , but as $\alpha$ grows the conditional expectation increasingly focuses on low realizations. ....	65
Figure 33. Pareto frontiers for different values of $\epsilon$ when solving the DRO model. The figure is analogous to Figure 32 for NPV-CVaR model. Note that this figure has a larger range of NPV values, but is otherwise similar to Figure 32. Note that we only include non-dominated solutions from Table 25. ....	75
Figure 34. Compares histograms for two values of epsilon, 0 (red) and 10 (blue).....	76
Figure 35. Probability mass functions for budget scenarios as risk aversion grows. ....	78
Figure 36. Probability mass functions for medium-risk project scenarios. ....	78
Figure 37. From ER data to decision making using PHM-RIAM models and methods. ....	82
Figure 38. Classification scheme for the considered maintenance approaches and the relative optimization methods developed in this report and in [2,3]. ....	83
Figure 39. Overview of the developed optimization methods. ....	84



## TABLES

Table 1. Case 1: cost and reliability data for the components of the syste shown in Figure 10. ....	23
Table 2. Case 1: system configurations lying on the Pareto frontier and satisfying the cost and unavailability constraints. ....	26
Table 3. Case 2: unavailability and cost data for each component of Figure 10. ....	27
Table 4. Case 2: system configurations lying in the Pareto frontier and satisfying the cost and unavailability constraints. ....	29
Table 5. Data for the maintenace optimization use case. ....	30
Table 6. Schedule options for each maintenance activity. ....	30
Table 7. Summary of the risk data for each maintenance activity. ....	31
Table 8. Optimal maintenance schedule. ....	33
Table 9. Summary of discrete optimization problems. ....	37
Table 10. RIAM application of discrete optimization problems. ....	37
Table 11. Multiple knapsack problem employed to test GA performances. ....	45
Table 12. Point estimates of the annual capital budget over a five-year planning horizon for the multiple knapsack problem test case. ....	45
Table 13. LOGOS solution for the multiple knapsack problem test case indicated in Table 11 and Table 12. ....	45
Table 14. Candidate set of 16 projects for an example problem in capital budgeting. The “Category” column distinguishes optional and must-do projects. The “NPV” column provides a point estimate of each project’s net present value. “Capital costs” provide point estimates of the liability streams induced by selecting each project over a five-year horizon. Values are in millions of dollars. ....	50
Table 15. Point estimates of the annual capital budget over a five-year planning horizon. Values are in millions of dollars. ....	51
Table 16. Optimal solution to the deterministic capital budgeting problem in which we select 4 of 10 optional projects, and all must-do projects. This solution respects the annual capital budgets of Table 12 and maximizes NPV, achieving a portfolio NPV of \$19.90M. ....	51
Table 17. The table replicates information from Table 14 except that most of the projects now have two or three implementation options via Plan A, B, and/or C. Values are in millions of dollars. ....	52
Table 18. We replace the point forecast of the annual capital budget over a five-year horizon in Table 12 with 10 scenarios. The probability distribution puts equal mass (probability 0.1) on each of the 10 values, and the values are perfectly correlated across time; e.g., if the budget realization is scenario 6 (S6) then the budget is \$31.0M in each of the five years. Values are in millions of dollars. ....	53
Table 19. Example solution under 10 budget scenarios, S1-S10. If a project has a “0” in the corresponding entry, that project is not selected under that scenario, and it is selected under entry “1.” The must-do projects are selected under all scenarios. The right-most	

column indicates the priority level associated with each project. For example, there are seven projected tied for the highest priority because they are selected under all scenarios, and there are two projects tied for priority 15-16 because they are not selected under any scenario. For simplicity of presentation, this example has no timing options..... 54

Table 20. The table replicates information from Table 14, and also shows the uncertain NPVs. The NPV values for the low-risk and medium-risk projects are shown for the three possible scenarios: pessimistic, most likely, and optimistic. The probabilities associated with the scenarios are given in Table 18. No-risk projects 2, 4, 5, 7, 9, 10, 12, and 14 are not subject NPV uncertainty and their values repeat those shown in Table 14..... 55

Table 21. The table shows no-risk, low-risk, and medium-risk projects. There are pessimistic, most likely, and optimistic scenarios for the NPVs of these projects, which are realized with the probabilities shown in the table. .... 56

Table 22. Optimal prioritization when both budget, costs, and NPV values are uncertain, with a total of 90 scenarios. The “Priority” column indicates the priority level, ranging from 1-16. The columns for Plan A, B, C, and Do Nothing indicate the number of scenarios in which each option is selected. Note that while “must do” projects are selected under all 90 scenarios, their priority can decrease based on the timing of their implementation. The expected NPV associated with this prioritization is \$168.90M. .... 57

Table 23. Optimal project prioritization. Here we optimize a weighted sum of the expected value of NPV (weight  $1-\lambda$ ) and CVaR (weight  $\lambda$ ). The respective NPV and CVaR values are as follows:  $\lambda=0$ : (NPV=168.90, CVaR=140.54),  $\lambda=0.5$ : (NPV=167.68, CVaR=142.72), and  $\lambda=1$ : (NPV=166.51, CVaR=143.05). We note that we write  $\lambda=0^+$  and  $\lambda=1^-$  in the column headers because we prefer to place small positive weight on CVaR in the former case and NPV in the latter case. For example, solving with  $\lambda=0.0$  yields the same NPV but CVaR of 128.16, a dramatic increase in risk, while using  $\lambda=0.01$  or 0.05 results in the same NPV but eliminates low NPV scenarios. .... 62

Table 24. Optimal project prioritization from solving model (1) when  $\alpha=0.75$  and  $\lambda$  ranges from 0.1 to 0.9. .... 66

Table 25. Optimal project prioritization from solving the distributionally robust optimization model when  $\epsilon$  ranges from 0 to 1000. This table can be compared with Table 24 when optimizing a weighted sum of NPV and CVaR..... 74

Table 26. Frequencies for plans A, B, and C and do nothing under the DRO approach for  $\epsilon = \mathbf{0}$  and  $\epsilon = \mathbf{10}$ . .... 77

Table 27. Emphases and timeframes for system health and asset management Use Cases..... 80

Table 28. List of steps for the combined PHM-RIAM workflow..... 82

## ACRONYMS

AM	Asset Management
BE	Basic Event
CDF	Core Damage Frequency
CVaR	Conditional Value at Risk
DKP	Multidimension Knapsack Problem
DOE	Department of Energy
DRO	Distributionally Robust Optimization
DUS	Distributional Uncertainty Set
EPRI	Electric Power Research Institute
ER	Equipment Reliability
ESF	Engineered Safety Features
ET	Event Tree
FT	Fault Tree
GA	Genetic Algorithm
HPI	High Pressure Injection
I&C	Instrumentation and Controls
ILCM	Integrated Life Cycle Management
INL	Idaho National Laboratory
INPO	Institute of Nuclear Power Operations
IPOP	Investments Portfolio Optimal Planning
IRR	Internal Rate of Return
JH	Jensen-Hughes
LCMP	Life Cycle Management Plan
LERF	Large Early Release Frequency
LTAM	Long-Term Asset Management
LWRS	Light Water Reactor Sustainability
MBSE	Model Based System Engineering
MC	Monte-Carlo
MCS	Minimal Cut Set
MKP	Multi-Knapsack Problem
MCKP	Multiple-choice Knapsack Problem

MMKP	Multidimension Multiple-choice Knapsack Problem
MSPI	Mitigating Systems Performance Index
MWe	Megawatt of Electrical power
MWh	Megawatt hour of energy
NAM	Nuclear Asset Management
NEI	Nuclear Energy Institute
NPP	Nuclear Power Plant
NPV	Net Present Value
O&M	Operations and Maintenance
PDF	Probability Density Function
PRA	Probabilistic Risk Assessment
PWR	Pressurized Water Reactor
RAVEN	Risk Analysis Virtual Environment
RCP	Reactor Coolant Pump
RIAM	Risk Informed Asset Management
RI-PSH	Risk Informed Plant System Health
RISA	Risk Informed Systems Analysis
RUL	Remaining Useful Life
RWST	Refueling Water Storage Tank
R&D	Research and Development
SA	Simulated Annealing
SKP	Single Knapsack Problem
SLR	Second License Renewal
SPSA	Simultaneous Perturbation Stochastic Approximation
SQA	Software Quality Assurance
SR <sup>2</sup> ML	Safety Risk and Reliability Model Library
SSCs	Structures, Systems, and Components
T&M	Testing and Maintenance
UQ	Uncertainty Quantification
VaR	Value-at-Risk

# Development and Application of a Risk Analysis Toolkit for Plant Resources Optimization

## 1. INTRODUCTION

This report summarizes the R&D activities of the Risk-Informed Asset Management (RIAM) project during fiscal year 2020 (FY-20) under the Risk Informed System Analysis (RISA) pathway for the Light Water Reactor Sustainability (LWRS) program [1]. This project focuses on the development of methods designed to optimize plant operations (e.g., maintenance and replacement schedule, optimal maintenance posture) provided system and component health and cost data.

In this respect, we are developing several classes of optimization methods that can be applied to a large variety of plant operation problems, i.e., Equipment Reliability (ER) and asset management related. This project development lives in cooperation with the Plant Health Management (PHM) [2] project, which focuses on the development of methods that integrate component health data and propagate this information at the system level to determine system sources of risk.

While the PHM project aims to assess current risk (e.g., safety or economic) related to a system, the RIAM project aims to reduce this risk or balance the risk with operational cost. The data and the models that are currently under development for the PHM project are, in fact, directly employed within the optimization methods developed in the RIAM project.

This year's activities for the RIAM project focused on the continuation of schedule optimization algorithms developed in FY-19. While in FY-19 we focused on both deterministic and stochastic capital budgeting methods [3], in FY-20 we moved forward by implementing two versions of schedule optimization methods. The first one reformulates the capital budgeting problem in a distributionally robust form [4] which allows the user to rely on data directly rather than proposing a distribution from the data itself. The second version reformulates the capital budgeting explicitly using risk measures as variables to maximize or minimize.

In parallel to these data-based optimization methods, we have also made progress in the development of model-based optimization methods in both continuous (e.g., gradient descent methods) and discrete form (e.g., genetic algorithms [5]). While data-based optimization methods are framed to a specific class of problems (e.g., multiple knap-sack problem), model-based optimization methods are independent from the class of problem under consideration. The user can in fact design fairly complex models that integrate reliability and cost models at the component level to determine system configurations that minimize costs provided risk constraints.

During FY-20, we also focused on the development of methods designed to identify the optimal maintenance posture based on the Pareto frontier analysis [6]. Rather than performing a tradeoff analysis (i.e., identify the absolute best posture), the Pareto frontier analysis performs a trade space exploration approach (i.e., identify value and costs of several postures and let the analyst impose desired value and cost constraints). This is performed by identifying maintenance postures that maximize value (e.g., system availability) and minimize operational costs, i.e., the Pareto frontier in a value-cost trade space.

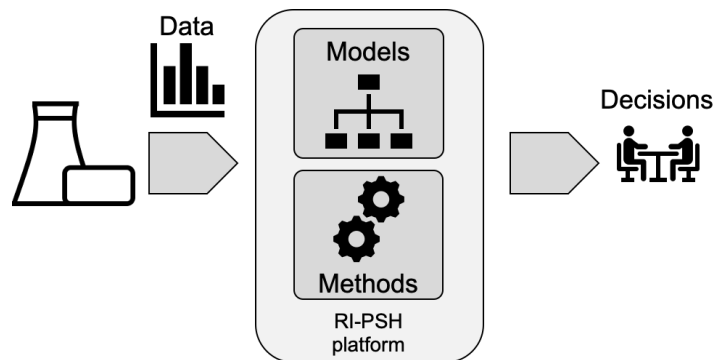
From a plant operation perspective, a Pareto frontier analysis is designed for “here-and-now” decisions (i.e., planning phase) while the developed optimization methods are designed for “wait-and-see” decisions (i.e., during operation).

In this report, we also show several applications of the developed methods and how they are linked to the PHM project. Lastly, we summarize how the development of these optimization methods has been structured with the vision of releasing them to the industry as open-source tools.

## 2. COST AND RISK CATEGORIZATION R&DD PATH

The goal of the Cost and Risk Categorization research path under the LWRS-RISA project [1] is to leverage advanced computational capabilities to support enhanced system performance and health management. The first objective of this effort is to integrate various elements of system health monitoring, management, and reporting in a manner that is significantly less labor intensive and is at least as technically effective as current programs. The second objective is to manage equipment and system performance and its financial risk and reduce costs associated with monitoring and regulatory compliance.

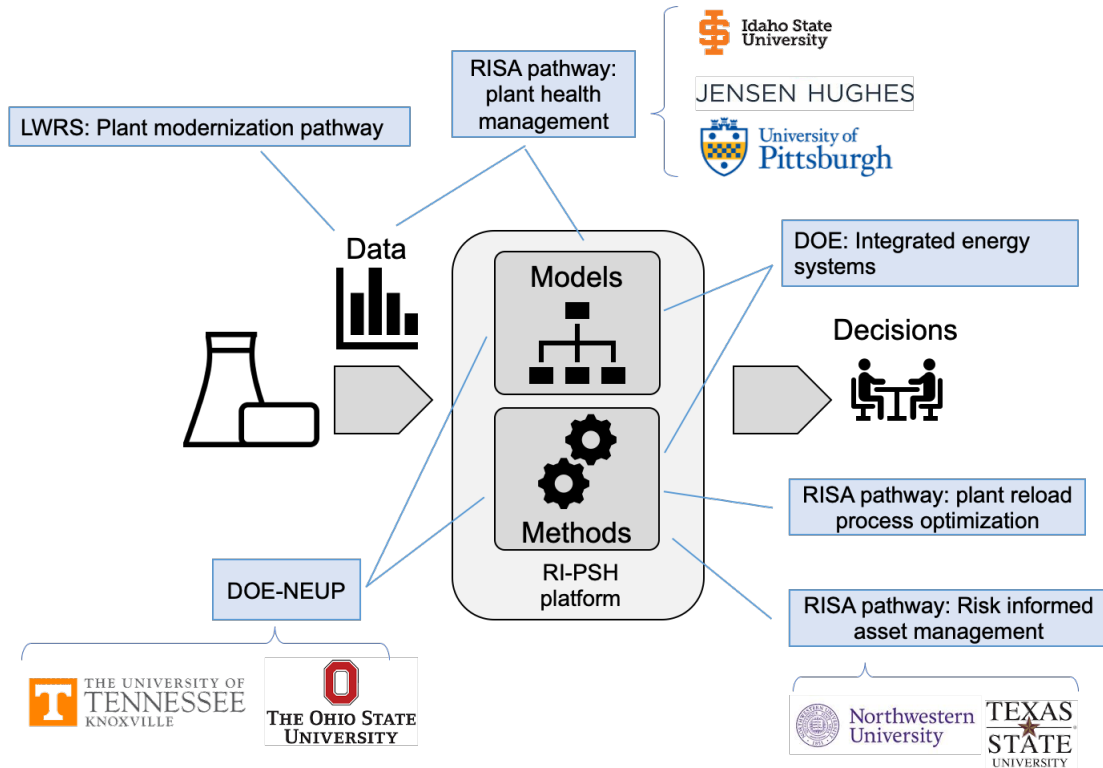
While the first objective is covered by the PHM project [2], the second objective is covered by the RIAM project. These two projects are coordinated to materialize the goal of the “Cost and Risk Categorization” research path into a software platform which is referred to here as Risk Informed Plant System Health (RI-PSH) platform (see Figure 1).



**Figure 1. High level description of the RI-PSH platform.**

The RI-PSH platform can be summarized as a Model Based System Engineering (MBSE) [7] platform for system operations: from system health data, it provides decision-making knowledge on the best maintenance posture and optimal component maintenance/replacement schedule. By maintenance posture we refer here as the maintenance strategy for each component that reduces O&M costs while maintaining adequate system availability. The term MBSE summarizes the main feature of the RI-PSH platform: rather than focusing on specific O&M applications, we are providing to the analyst a set of models and methods along with a computational analysis framework. Depending on the issue to be analyzed, the analyst can then assemble models together (e.g., a combination of reliability and cost models), and apply to it a series of computational methods (e.g., optimization, uncertainty quantification, and data analysis).

From a development perspective, the PHM project is focusing mainly on the development of models and health data integration while the RIAM project is focusing on the development of optimization methods. During this and past years, this development has been coordinated with other projects with another LWRS pathway (the plant modernization pathway) and other DOE-NEUP projects. Figure 2 shows a complete picture of the external collaborators that contribute effectively to the PHM-RIAM projects. These collaborators include both academic and industry partners.



**Figure 2. Interactions among the RIAM project and other DOE projects in the development of the RI-PSH platform.**

We have structured the RI-PSH platform on different repositories where each repository contains a specific class of models or methods. Figure 3 shows the hierarchical structure of the repositories currently being developed and maintained. The items shown in yellow represent those fully developed under the RIAM-PHM projects while the items shown in red are co-developed with other programs. A more detailed description of these repositories is presented as follows:

- **RAVEN [8]:** It is a flexible and multi-purpose uncertainty quantification, regression analysis, PRA, data analysis and model optimization software. Depending on the tasks to be accomplished and on the probabilistic characterization of the problem, RAVEN perturbs (through Monte-Carlo, Latin Hypercube, reliability surface search [8] sampling methods) the response of the system by altering its parameters. The data generated by the sampling process is analyzed using classical and more advanced data mining approaches. RAVEN also manages the parallel dispatching (i.e., both on desktops, workstations and large high-performance computing machines) of the software representing the physical model. RAVEN heavily relies on artificial intelligence algorithms to construct surrogate models of complex physical systems in order to perform UQ, reliability analysis (limit state surface), and parametric studies.
- **LOGOS:** It contains a set of discrete optimization models that can be employed for capital budgeting optimization problems, and LOGOS integrates economic and reliability risk into a single analysis framework. More specifically, provided systems, SSCs health (e.g., failure rate or failure probability), O&M costs, replacement costs, cost associated to component failure and budget constraints, LOGOS provides the optimal set of projects (e.g., SSC replacement) that maximizes profit and satisfies the provided requirements. The items listed above have input data that can be either deterministic or stochastic in nature, i.e., they can be point values or probability distribution functions. In the latter case, several scenarios are generated by sampling the provided distributions.

The developed models are based on different versions of the knapsack optimization problem. Two main classes of optimization models have been initially developed: deterministic and stochastic. Stochastic optimization models evolve deterministic models by explicitly considering data uncertainties (associated to constraints or item cost and reward). These models can be employed as stand-alone models or interfaced with the INL developed RAVEN code to propagate data uncertainties and analyze the generated data (i.e., sensitivity analysis).

- **SR<sup>2</sup>ML:** It is a software package which contains a set of reliability models designed to be interfaced with the INL developed RAVEN code. These models can be employed to perform both static and dynamic system risk analysis and determine risk importance of specific elements of the considered system. Two classes of reliability models have been developed; the first class includes all classical reliability models (fault trees, event trees, Markov models and reliability block diagrams) which have been extended to deal not only with Boolean logic values but also time dependent values. The second class includes several component ageing models. Models included in these two classes are designed to be included in a RAVEN ensemble model to perform time dependent system reliability analysis (dynamic analysis). Similarly, these models can be interfaced with system analysis codes within RAVEN to determine failure time of systems and evaluate accident progression (static analysis).
- **TEAL:** It enables the capability to compute the NPV (Net Present Value), the IRR (Internal Rate of Return) and the PI (Profitability Index) with RAVEN. Furthermore, it is possible to do an NPV, IRR or PI search, i.e. TEAL will compute a multiplicative value (for example the production cost) so that the NPV, IRR or PI has a desired value. The plugin allows for a generic definition of cash flows which drivers are provided by RAVEN. Furthermore, TEAL includes flexible options to deal with taxes, inflation, discounting and offers capabilities to compute a combined cash flow for components with different component lives.
- **VERT:** The Versatile Economic Risk Tool (VERT) is a model library designed to perform Generation Risk Assessment (GRA). VERT quickly and effectively evaluates the economic risk systems and sub-systems impose on NPPs. This is performed by employing classical PRA tools, such as Fault Trees, with component reliability and availability models to evaluate risk associated to loss of production.
- **SRAW:** The System Risk Analysis Workflow (SRAW) plugin for the RAVEN code. SRAW, as a plugin, enables RAVEN to perform stochastic analysis of NPP asset management. The primary function of SRAW is to generate the complex RAVEN workflows necessary to optimize asset management under various scenarios.

While some repositories are already available with an open-source license (RAVEN and TEAL), the vision is to release all of them with an open-source license within FY-21. This will greatly improve collaboration and development with external collaborators.

### 3. METHODS FOR PLANT RESOURCES MANAGEMENT

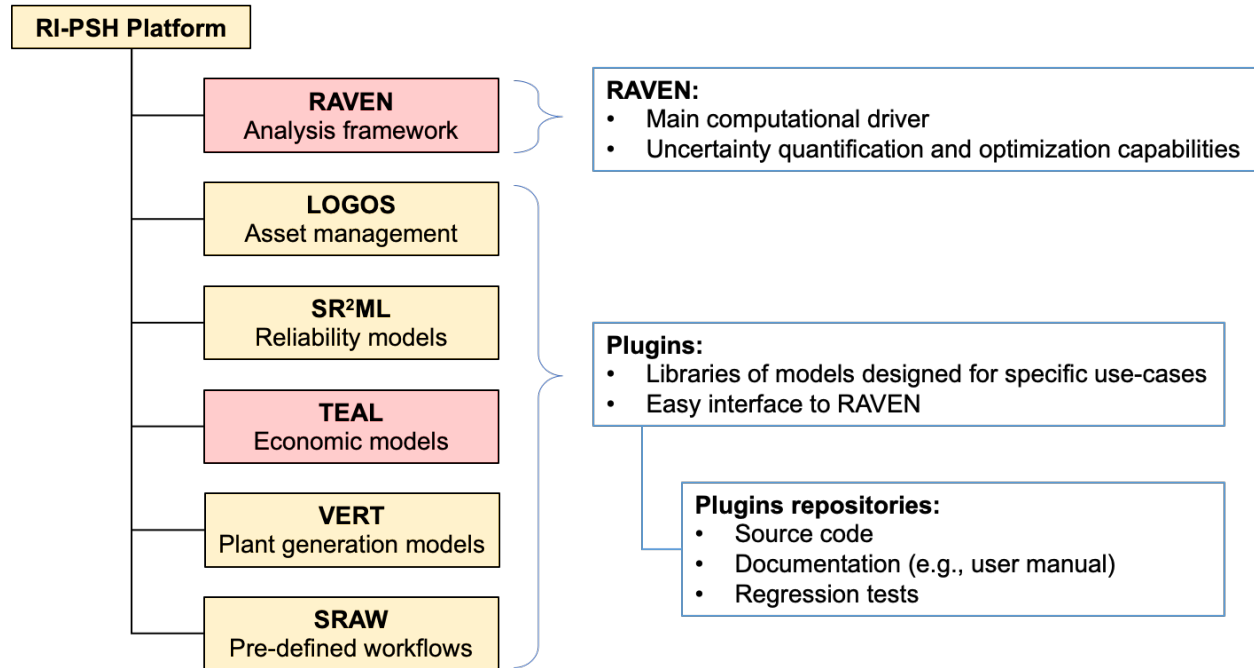
As mentioned in the introduction, the RIAM project is focusing on the development of methods designed to optimize plant resources (e.g., SSC, personnel, and ER activities). These methods can be classified as either:

- Sampling methods
- Optimization methods.

Sampling methods are designed mainly to propagate data and model uncertainties (e.g., investment evaluation). Optimization methods are designed to determine the best solution to a problem that satisfies a



limited set of criteria (also known as constraints). In this report we will briefly describe the sampling methods while we will concentrate more on the optimization methods developed during FY-20 and highlight how they relate to the ones developed during FY-19.



**Figure 3. Structure of the RI-PSH computational platform.**

Depending on the problem to be solved and the type of data available, the optimization algorithm to be used might change. As an example, the data structure might be either discrete or continuous in nature. In addition, the problem under consideration might require a specific data set or, alternatively, a specific model (which balances system reliability/availability and cost) which changes the problem structure depending on the considered boundary conditions.

Figure 4 shows in a graphical form the available algorithms developed within the RIAM project for plant resources management depending on the data structure and the method that the problem requires. In the following sections we will cover all these methods. Note that some of these methods have been developed during FY-19 (and they will be briefly presented) while several others have been developed during FY-20 (and they will be extensively described).

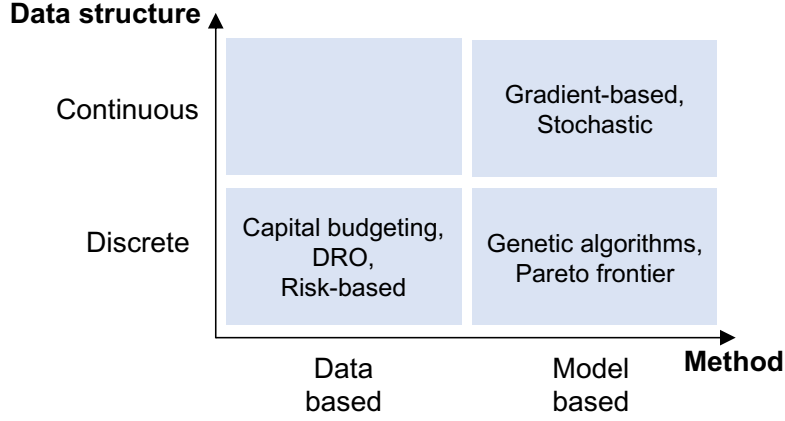
## 4. INVESTMENT EVALUATION

The first application of the RIAM tool set can be the evaluation of the impact of an investment from an economic perspective. In [2] we have presented a mathematical formulation to measure economic impact of an investment that directly affects the reliability and availability of the plant or system.

It is here assumed that:

- A single system  $sys$  is analyzed
- The considered system is composed of a set of  $N$  components

- System failure can be uniquely determined from the logical status of its components (e.g., by employing a Fault-Tree logic structure)
- Time horizon is fixed (i.e.,  $[0, T^{Max}]$ ) and it is discretized into  $T$  time intervals having identical length  $\Delta t$  (i.e.,  $T^{Max} = T \cdot \Delta t$ ).



**Figure 4. Classification of the optimization algorithms developed under the RIAM project based on employed data structure and the method being used.**

This formulation is based on the evaluation of the NPV for the operation of the system (here indicates as  $NPV^{operation}$ ) which is a mathematical balance between:

- $PV^{operation}$ : the present value of plant or system operation
- $PV^{life}$ : the present value of plant or system life costs
- $PV^{failure}$ : the present value of plant or system failure.

The mathematical formulation for  $NPV^{operation}$  is discretized in time interval and it is shown as follows:

$$\begin{aligned}
 NPV^{operation} &= PV^{operation} - PV^{life} - PV^{failure} = \\
 &= \sum_{t=1}^T R_{t-1}(sys) \frac{(1 - p_t(sys)) \cdot V_t}{(1 + ir)^t} \\
 &\quad - \left( \sum_{n=1}^N C_n^{proc} + \sum_{n=1}^N \sum_{t=1}^T \frac{C_{t,n}^{OM}}{(1 + ir)^t} \right) \\
 &\quad - \sum_{t=1}^T R_{t-1}(sys) \frac{\sum_{n=1}^N p_t(n) \cdot p_t(sys|n) \cdot C_{t,n}^{failure}}{(1 + ir)^t}
 \end{aligned} \tag{1}$$

where:

- $R_{t-1}(sys)$ : reliability of the considered system  $sys$  within time interval  $t - 1$
- $p_t(sys)$ : probability of failure of the considered system  $sys$  within time interval  $t$

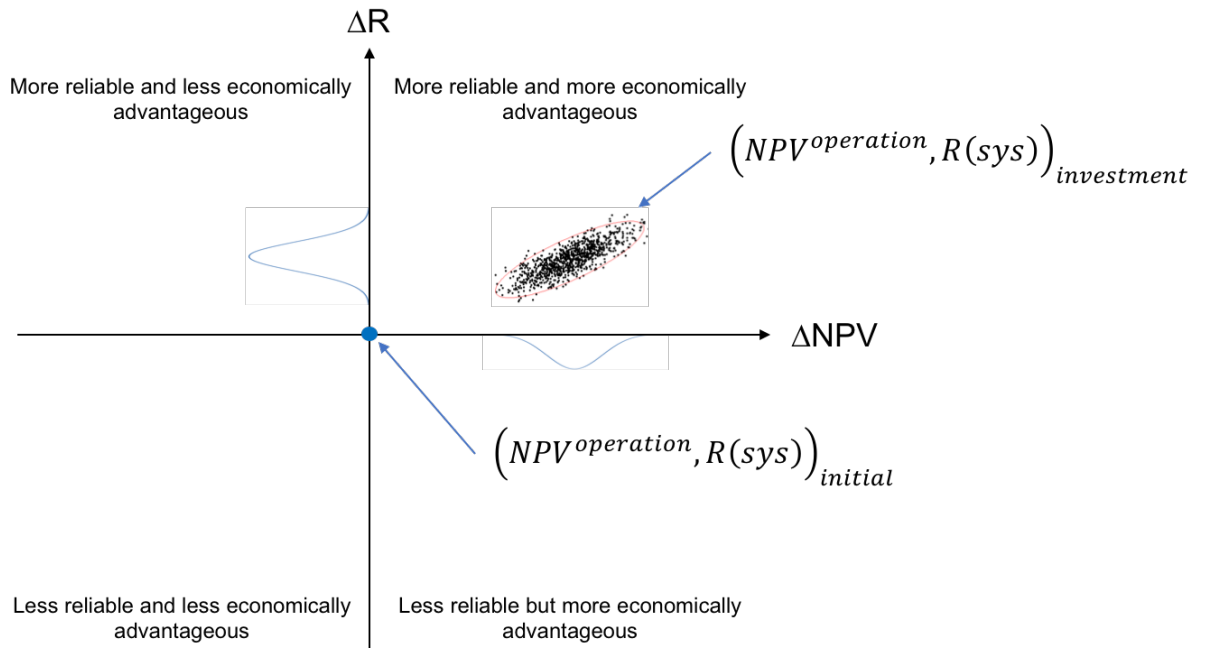
- $V_t$ : economic production value (e.g., power generation) due to the correct operation of the system within time interval  $t$
- $C_n^{proc}$ : procurement costs of component  $n$  (note that here more complex supply chain models can be added)
- $C_{t,n}^{OM}$ : O&M costs for component  $n$  within time interval  $t$
- $p_t(n)$ : probability of failure of component  $n$  within time interval  $t$
- $p_t(sys|n)$ : probability of failure of the considered system  $sys$  within time interval  $t$  given that component  $n$  has failed
- $C_{t,n}^{failure}$ : cost associated with system failure caused by failure of component  $n$  (e.g., loss of production, replacement costs, or regulatory burden) within time interval  $t$ .

From a decision-making perspective, two possible paths can be followed:

- Evaluate and compare the actual  $NPV^{operation}$  with the  $NPV^{operation}$  generated by the investment
- Evaluate and compare  $(NPV^{operation}, R(sys))$  for both the actual and the investment.

An important factor to consider is that some elements in Equation (1) might not be certain but they might be affected by uncertainties. Thus, it is required to propagate uncertainties throughout the model. Graphically this can be plotted in a 2-dimensional graph (see Figure 5) where each dimension corresponds to the incremental  $NPV^{operation}$  and incremental  $R(sys)$ . Note that these two dimensions are correlated since  $NPV^{operation}$  is also function of  $R(sys)$ .

Another important feature about propagating uncertainties is that it is possible to evaluate the sensitivity of  $NPV^{operation}$  and  $R(sys)$  to a variation of each uncertain parameter. This sensitivity analysis might prove to be useful to rank the most relevant uncertain parameters from a decision-making standpoint.



**Figure 5. Reliability vs. NPV plot for evaluation of candidate projects [19].**

## 5. PARETO FRONTIER ANALYSIS

The Multi-Attribute Utility Analysis (MAUA) is a structured methodology in decision theory designed to handle the tradeoffs among multiple objectives [9]. Provided several options, the goal is to identify the “best” option that satisfies a specific set of needs or requirements.

The first step is to identify the set of attributes that affect the decision at hand. Typically, these attributes can be condensed into two, utility and cost, but, in some applications, the number of attributes might be higher (e.g., other factors might be: lifecycle cost, performance, etc.).

Let’s assume that a decision can be taken from a set of options by considering the utility and cost of each option. Using a graphical representation (see Figure 6), it is possible to plot each option as a point in a 2-dimensional space, cost vs. utility<sup>a</sup>:

- *Cost*: this axis represents the cost associated with each option ranging from 0 (i.e., cheapest option) to a maximum value  $C_{max}$  (i.e., the most expensive option)
- *Utility*: this axis represents the added value (or the performance) associated with each option ranging from 0 (i.e., lowest performance option) to a maximum value  $U_{max}$  (i.e., option with highest performance).

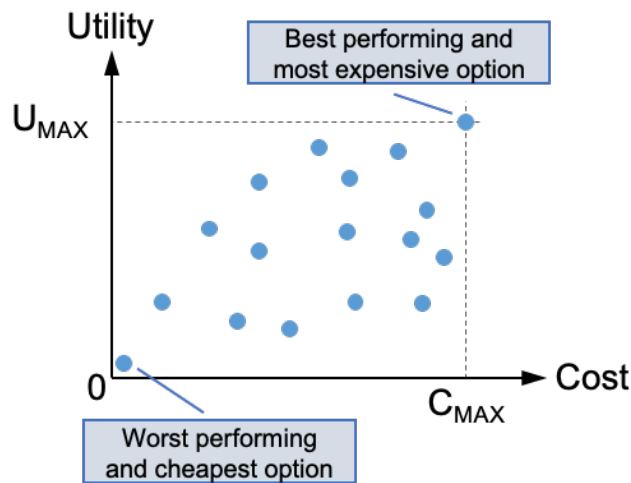


Figure 6. Set of options (blue dots) plotted in a cost vs. utility space.

Once the complete set of options have been generated and the utility and cost values have been determined for each option, the next step is the determination of the Pareto optimal frontier which is fundamentally an envelope of options that dominates (in terms of both utility and cost) the set of remaining options (see Figure 7).

The final step in the analysis is to impose the utility and cost constraints (see Figure 8) and select those points that satisfy both of these requirements. In some applications, the data provided to each option to generate its values of utility and cost might be affected by uncertainties. In such cases, data uncertainties are propagated for each selected option; the final outcome is that rather than having a single point for each option, we have a cloud of options centered around the selected option. By performing this analysis, now it is possible to determine not only the best options, but also how uncertainties affect such a decision process.

---

<sup>a</sup> As indicated earlier, the number of attributes considered in complex settings can be  $N > 2$ . Thus, in such cases, the space would be  $N$  -dimensional.

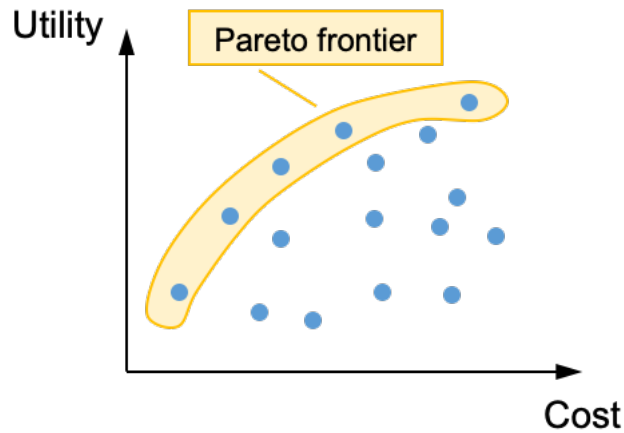


Figure 7. Pareto frontier obtained from a set of options plotted in a cost vs. utility space.

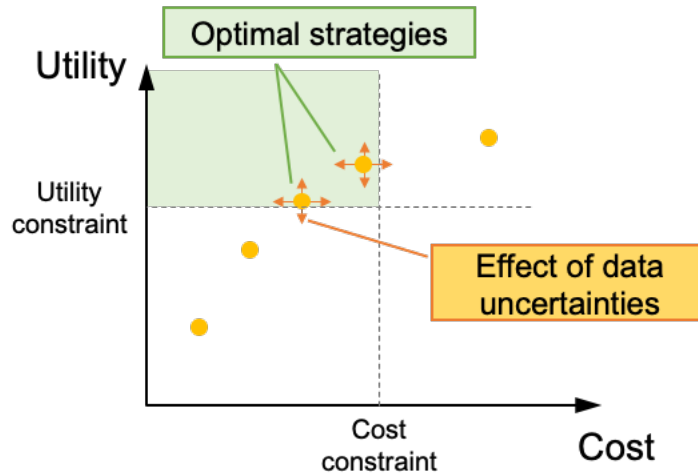


Figure 8. Propagation of uncertainties for the points on Pareto frontier and imposition of cost and utility constraints.

## 5.1 Pareto Frontier Analysis for RIAM Applications

Within the scope of the PHM and RIAM project, a Pareto frontier analysis can mainly be used for two kinds of use cases:

- Identification of plant or system optimal maintenance posture
- Identification of the optimal set of plant or system maintenance operations

For simplicity, let's consider a system  $sys$  composed by a set of  $N$  components  $C_n$  ( $n = 1, \dots, N$ ).

### 5.1.1 Identification of Plant/System Optimal Maintenance Posture

In the first use case, assuming a failure mode for each component, a set of  $R_n$  possible maintenance strategies can be chosen for each component  $n$  ( $n = 1, \dots, N$ ). Let's indicate with  $M_n^r$  ( $r = 1, \dots, R_n$ ), a maintenance strategy for component  $n$ . For each  $M_n^r$  it is possible to determine component availability  $A_n^r$  (or, alternatively, reliability) and the corresponding maintenance costs  $C_n^r$ .

Provided a model  $\mathcal{A}$  which relates system availability  $A_{sys}$  to component availabilities  $A_n^r$ , and a model  $\mathcal{C}$  which relates system cost  $C_{sys}$  to component costs  $C_n^r$ , it is possible to define a maintenance posture option  $\mathcal{M}$  as the set of maintenance strategy chosen for each component:

$$\mathcal{M} = (M_1, \dots, M_n, \dots, M_N)$$

where  $M_n$  is the maintenance option chosen for component  $n$ :  $M_n = (A_n, C_n)$ ,  $C_n \in \{C_n^r\}_{r=1}^{R_n}$ ,  $M_n \in \{M_n^r\}_{r=1}^{R_n}$

For each maintenance posture option  $\mathcal{M}$ , it is possible to generate the corresponding values of  $A_{sys}$  and  $C_{sys}$  for that option:

$$\begin{aligned} A_{sys} &= \mathcal{A}(A_1, \dots, A_N) \\ C_{sys} &= \mathcal{C}(C_1, \dots, C_N) \end{aligned} \quad (2)$$

Assuming independence between maintenance strategies  $M_n^r$ , the complete number of maintenance posture options can be calculated as:

$$\prod_{n=1}^N R_n \quad (3)$$

At this point, optimal maintenance posture can be determined using the Pareto frontier analysis as indicated in Section 5.1. In this case utility is substituted by system availability while cost is substituted by system maintenance cost (see Figure 9). Each point in this 2-dimensional space is characterized by specific values of  $A_{sys}$  and  $C_{sys}$ .

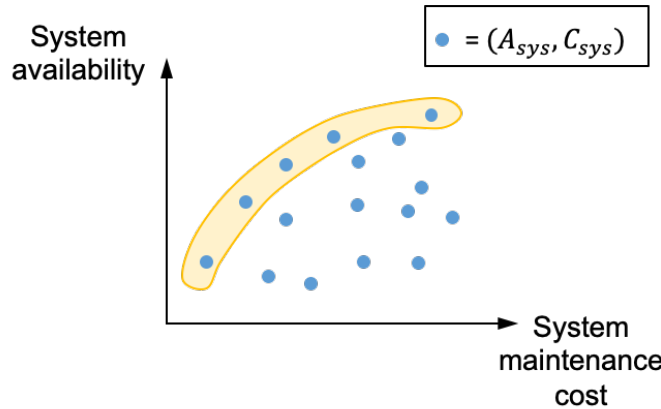


Figure 9. Pareto frontier analysis in an availability vs. cost space.

### 5.1.2 Identification of the Optimal Set of Plant/System Maintenance Operations

As indicated previously, the primary objective of a plant maintenance program is to effectively and efficiently maintain plant SSCs so that plant safety and production are maximized in a manner that is cost effective throughout the expected operational lifetime of the plant (including periods of license renewal). Although the performance of the current fleet of operating NPPs has achieved exemplary levels of performance in terms of safety and power production, consistently high maintenance expenditures, combined with implementation of safety upgrades required as a result of the Fukushima Daiichi accident, have challenged the economic viability of these plants.

The approach described in previous sections is intended to provide an integrated system and suite of tools to support plant decision makers to evaluate the impact of plant maintenance decisions on critical parameters important to plant safety and power production (i.e., system availability and reliability) while minimizing both short- and long-term maintenance costs. The following provides an outline of how this system would be implemented and used.

1. Identify various maintenance options for each component; for purposes of illustration assume  $r_n$  options are possible for component  $n$ . These options can be developed from a number of existing sources including the following:
  - Current assigned maintenance activities (including use of existing SSC maintenance templates)
  - Application of “alternative treatments” specified in a plant’s 10CFR50.69 program (if applicable)
  - Additional information from industry experience and vendor or manufacturer recommendations.

2. Define availability and cost model for each component option:

$$(A_n, C_n)_r \quad r = 1, \dots, R_n$$

In this step, it is required to evaluate the impact of each maintenance option from both availability and cost perspective at the component level. Existing sources might include the following:

- ER data for similar components from industry databases
  - Estimates of testing and maintenance costs based on industry practices (sources: EPRI, NEI databases).
3. For all possible option combinations, evaluate system availability and system operational cost
  4. Plot the evaluations of Step 3 in a 2-dimensional space (availability vs. cost), (e.g., see conceptual example in Figure 6)
  5. Evaluate Pareto frontier for the plot of Step 4 (see Figure 7) to identify the most cost effective combinations of maintenance strategies to achieve desired plant safety and operational (e.g. production) objectives
  6. Propagate uncertainties for the points located on the Pareto frontier (see Figure 8)
  7. Impose cost and utility constraints (see Figure 8)
  8. Select best candidate point (decision maker).

## 5.2 Pareto Frontier Post-Processor

Pareto frontier analysis can be performed in RAVEN using the `ParetoFrontier` post processor. This post processor is designed to identify the points lying on the Pareto frontier in a cost-utility space. This post processor receives as input a dataset which contains all data points in the cost-utility space and it returns the subset of points lying in the Pareto frontier as a new dataset. The generation process of the input dataset is shown in the set of provided examples (see Section 6).

The following algorithm provides the main operations for evaluating the Pareto frontier (cost is indicated as  $C$  and utility as  $U$ ):

1. Project all points on cost dimension in ascending order, e.g.  $P_1, \dots, P_i, P_{i+1}, \dots, P_{max}$
2. Add the first point (i.e., less expensive point) to the set of the Pareto frontier (PF), i.e.,  $PF = [P_1]$
3. Move to the next point  $P_2$ , if  $U(P_2) > U(PF[-1])$ , then add  $P_2$  to the PF, i.e.,  $PF = [P_1, P_2]$ , else disregard  $P_2$  (as this point represents a nonoptimal maintenance strategy)
4. Repeat Steps 3 for all the rest points, i.e.,  $P_3, \dots, P_{max}$

## 5.3 Pareto Frontier Test Cases

This section provides practical examples on the application of the Pareto frontier analysis applied to system reliability evaluation, and maintenance optimization. Consider a Pressurized Water Reactor (PWR) High Pressure Injection (HPI) system. As shown in Figure 10, the HPI system consists of a set of components (mainly valves and pumps) that are required to maintain redundancy and increase system reliability.

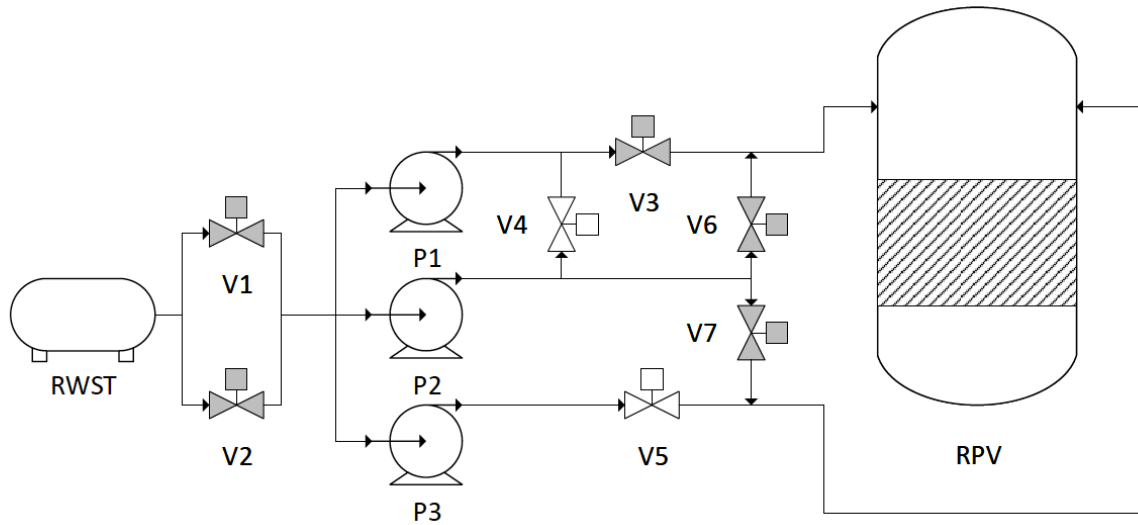


Figure 10. Graphical representation of the considered HPI system.



### 5.3.1 Case 1: System Reliability Modeling

The first application of the Pareto frontier analysis is dealing with the design of a system which aims to balance system cost and reliability. In this respect, it is assumed here that each component (i.e., valves and pumps) can be chosen from three options (i.e., A, B, and C).

For each option for both components, the reliability and cost data are reported in Table 1 (note that the option for each component does not have to be identical). The goal of the analysis is to determine the optimal design of the system that minimizes costs and maintains adequate system reliability/availability. (Note that since the HPI system is a standby system, both availability and reliability objectives will be applicable. However, for the purposes of illustration of the method we focus on the availability metric.)

**Table 1. Case 1: cost and reliability data for the components of the system shown in Figure 10.**

Component	Option	Failure rate	Cost
Valve	A	1.0 E-8 h <sup>-1</sup>	\$ 40,000
	B	5.0 E-7 h <sup>-1</sup>	\$ 30,000
	C	1.0 E-7 h <sup>-1</sup>	\$ 20,000
Pump	A	5.0 E-8 h <sup>-1</sup>	\$ 80,000
	B	1.0 E-7 h <sup>-1</sup>	\$ 70,000
	C	3.0 E-7 h <sup>-1</sup>	\$ 60,000

The analysis is structured in three steps:

- Evaluate system cost and reliability for all possible system configurations
- Determine the Pareto frontier in the unavailability vs. costs space at the system level
- Apply unavailability and costs constraints to points lying on the Pareto frontier.

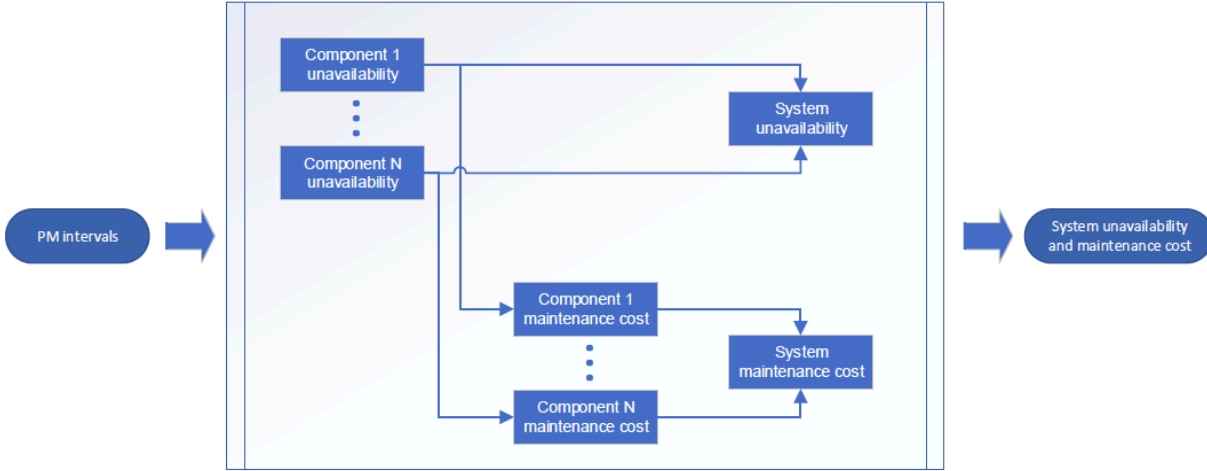
#### 5.3.1.1 Step 1

Given that we are dealing with ten components and each component can be chosen from three options, there are  $3^{10} = 59,049$  possible system configurations to evaluate. This step has been solved using the RAVEN [8] statistical framework by evaluating system cost and reliability for all possible system configurations.

System configuration cost and unavailability have been determined by employing:

- a) A RAVEN logic model for each component which provides component failure rate, reliability and unavailability values based on the sampled option
- b) Ageing models for each component which determine component unavailability and reliability provided the chosen failure rates provided in a)
- c) Cost models for each component based on the ageing models constructed in b)
- d) A system reliability model which determines system reliability provided component reliability values determined in b)
- e) A system cost model which determines system cost provided component cost values determined in c)

- f) A RAVEN ensemble model (see Figure 11) which is actually responsible to logically connect all the models shown in a) through e). This model receives input for the option chosen for each component and it provides the corresponding values of system cost and reliability.



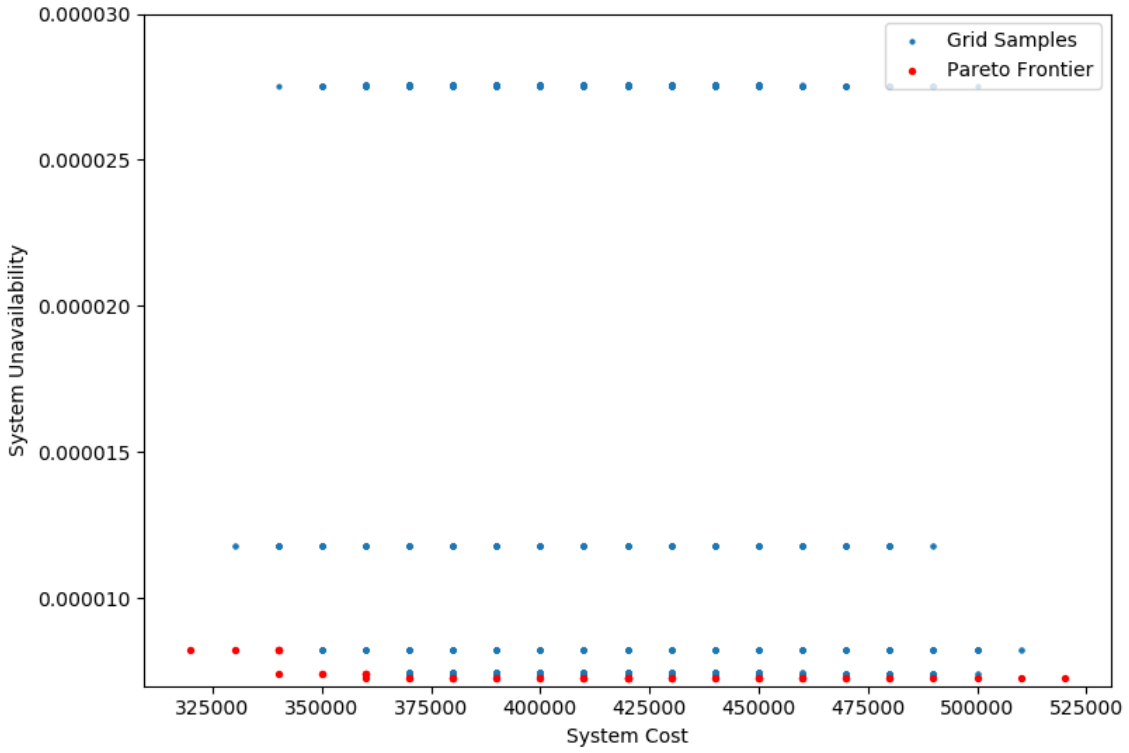
**Figure 11. Graphical representation of the RAVEN EnsembleModel for a generic system maintenance scheduling optimization problem.**

The model described above has been evaluated  $3^{10} = 59,049$  times by sampling the option for each of the ten components in order to determine system cost and reliability for all possible system configurations. Note that if the number of options and if the number of components grow, then the number of possible system configurations grows exponentially. This might be an issue if the computational time to evaluate system reliability is relevant (minutes rather than a fraction of a second). In this case, rather than using a PRA code such as SPAHIRE [10] or CAFTA [11] (which would increase computational time) we are employing an internal RAVEN model, the MCSsolver, which is designed to determine the probability associated with a Top Event (e.g., of a fault tree) provided the list of Minimal Cut Sets (MCSs) and the probability associated with each Basic Event (BE). The list of MCS is provided through a .csv file and is compatible with most existing PRA codes.

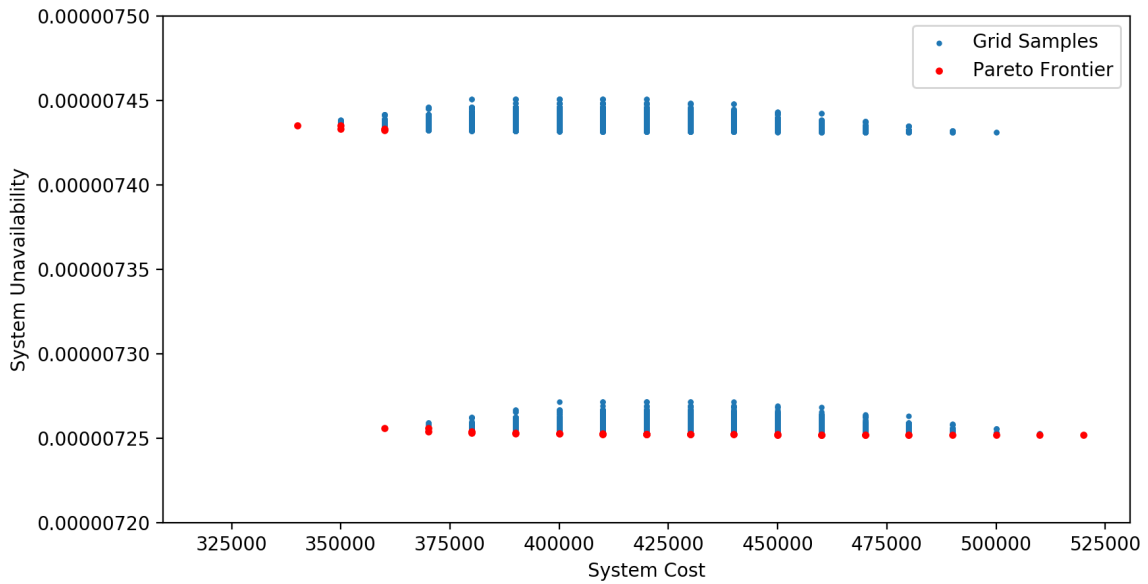
### 5.3.1.2 Step 2

Figure 12 plots the obtained values of system cost and reliability for all 59,049 possible system configurations (blue color). Note that the 2-dimensional space does not match exactly what has been presented in Section 5, i.e., we are dealing with a cost-unavailability space rather than a cost-value space. This is not an issue for the determination of the Pareto frontier, the only difference is that the layout of the Pareto frontier will differ from the one presented in Section 5.

Figure 12 also plots the points lying on the Pareto frontier (red points); as indicated earlier, the envelope of the Pareto frontier is slightly different from the one presented in Section 5. As expected, as system unavailability decreases, system cost increases. The determination of the Pareto frontier was performed within RAVEN using the ParetoFrontier post processor. Provided the data set generated in Step 1, this post processor generates a new data set which contains the points in the original data set that are within the Pareto frontier. Figure 13 provides a more detailed zoom of the of the Pareto frontier to explore the underlined structure of the points lying in the two lower clusters.



**Figure 12. Pareto frontier obtained for case 1.**



**Figure 13. Zoom of the Pareto frontier obtained for case 1.**

### 5.3.1.3 Step 3

Using the ParetoFrontier post processor used in Step 2, it is possible to specify the desired limits for both dimensions of the space under consideration (i.e., cost and unavailability). For this specific case we have set these limits to be:

- Max cost: \$ 400k
- Max unavailability: 7.5 E-5

If system cost and unavailability requirements are imposed on the set of points lying on the Pareto frontier, then it is possible to obtain 14 system configurations that satisfy these requirements as indicated in Table 2. Note that in these 14 system configurations, the resulting options chosen for the majority of the valves is identical (i.e., V1, V4, V5, V6, V7). Greater variability involves the choice for the pumps which mostly range between options B and C.

At this point the decision maker can chose the best solution by prioritizing either system cost or unavailability. In this example, the system unavailability does not change much among the 14 system configurations; however, system costs have a higher degree of variability (i.e., \$ 340K-390K). Given this situation, the most appropriate course of action (assuming there are no other considerations which could influence the decision) would be to choose system configuration #12: cheaper solution which satisfies the imposed unavailability constraints.

**Table 2. Case 1: system configurations lying on the Pareto frontier and satisfying the cost and unavailability constraints.**

#	V1	V2	V3	V4	V5	V6	V7	P1	P2	P3	Cost	Unav.
12	A	A	C	C	C	C	C	C	B	C	340K	7.4E-6
13	A	A	C	C	C	C	C	B	B	C	350K	7.4E-6
14	A	A	C	C	C	C	C	C	C	C	350K	7.4E-6
15	A	A	C	C	C	C	C	B	C	C	360K	7.4E-6
16	A	A	C	C	C	C	C	C	A	C	360K	7.4E-6
17	A	A	C	C	C	C	C	C	B	B	360K	7.4E-6
18	A	B	C	C	C	C	C	C	C	C	360K	7.3E-6
19	A	B	C	C	C	C	C	B	C	C	370K	7.3E-6
20	A	B	C	C	C	C	C	C	B	C	370K	7.3E-6
21	A	B	C	C	C	C	C	B	B	C	380K	7.3E-6
22	A	B	C	C	C	C	C	C	A	C	380K	7.3E-6
23	A	B	C	C	C	C	C	C	B	B	380K	7.3E-6
24	A	B	C	C	C	C	C	B	B	B	390K	7.3E-6
25	A	B	C	C	C	C	C	C	A	B	390K	7.3E-6

### 5.3.2 Case 2: Maintenance Reliability Modeling

The second application of the Pareto frontier analysis involves the determination of the optimal maintenance posture for the system shown in Figure 10. It is assumed here that all the components of the system of Figure 10 are subject to preventive maintenance. The time  $T^{PM}$  between two consecutive maintenance operations must be chosen for each component. For this example, three options are assumed to be available: 4,000 hours, 7,000 hours, and 10,000 hours.

Each component has been modeled using two models (see [2])

- Component unavailability model:

$$u = p_i^{PM} + \frac{T_i^{DT}}{T_i^{PM}} + P^F \quad (4)$$

where:

- $p_i^{PM}$ : Component unavailability due to PM error of omission
  - $\frac{T_i^{DT}}{T_i^{PM}}$ : Component unavailability due to performance of the PM activities
  - $P^F$ : Component unavailability due to component failure, i.e., the probability of failure of the component in  $[0, T_i^{PM}]$ .
- Component cost model:

$$\overline{Cost}_i^{Maint} = \frac{Cost_i^{PM} + P^F \cdot Cost_i^{CM}}{T_i^{PM}} \quad (5)$$

where:

- $Cost_i^{PM}$ : cost for performing PM
- $Cost_i^{CM}$ : cost for performing CM due to component failure.

Provided the cost and reliability values for the variable indicated above it is possible to determine component unavailability and maintenance cost for each option for each component as indicated in Table 3.

**Table 3. Case 2: unavailability and cost data for each component of Figure 10.**

Component	Option	$T^{PM}$ [hr]	Unavailability	Maintenance cost [\$/hr]
Valve	A	4,000	0.0198	0.8806
	B	7,000	0.0344	0.6479
	C	10,000	0.0595	0.6044
Pump	A	4,000	0.0302	0.1614
	B	7,000	0.0418	0.1626
	C	10,000	0.0654	0.1881

We then proceeded to construct a system model able to determine system reliability and system cost provided the chosen  $T^{PM}$  for each component. We have employed the following models available in the SR<sup>2</sup>ML repository:

- Component models:

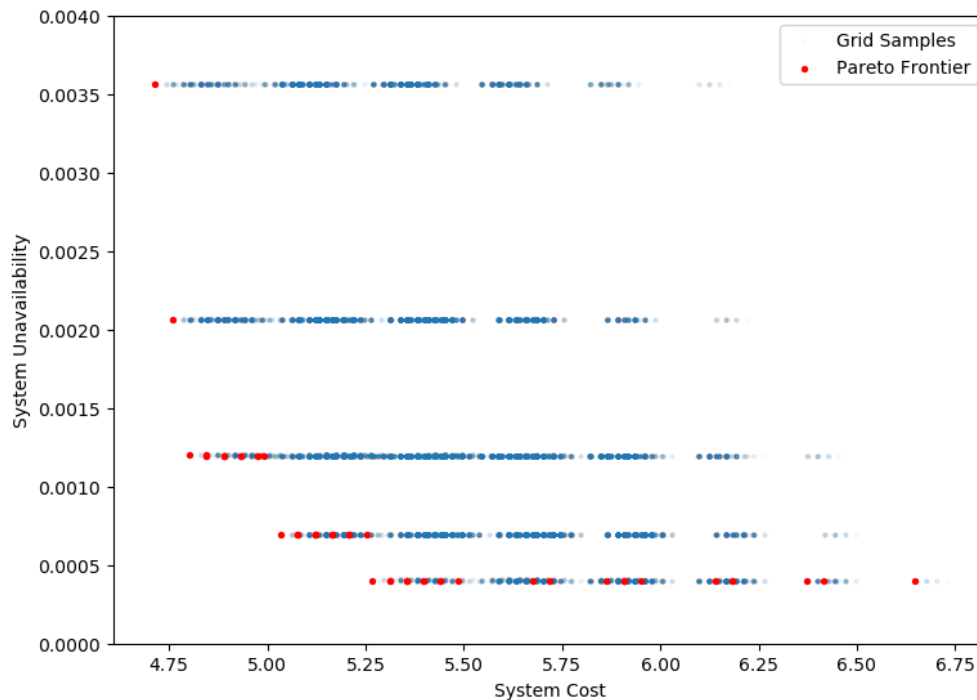
- Unavailability model
- Costs model.
- System models
  - Unavailability model
  - Costs model.

We have then employed the `EnsembleModel` feature of RAVEN (see [3]) to link all these models together in a single model as shown in Figure 11. The analysis has been completed by performing the following two steps:

- a) Identify the Pareto frontier in the unavailability vs. costs space at the system level
- b) Apply unavailability and costs constraints to point selected in Step a)

The model described above has been evaluated  $3^{10} = 59,049$  times by sampling the option for each of the ten components in order to determine system cost and reliability for all possible system configurations.

Figure 14 plots the obtained values of system cost and reliability for all 59,049 possible system configurations (blue color). Figure 14 also plots the points lying on the Pareto frontier (red points); as indicated earlier, the envelope of the Pareto frontier is slightly different from the one presented in Section 5. As expected, as system unavailability decreases, system cost increases. The determination of the Pareto frontier was performed within RAVEN using the `ParetoFrontier` post processor. Provided the data set generated in Step 1, this post processor generates a new data set which contains the points in the original data set that are within the Pareto frontier. In this case the Pareto frontier contains 64 system configurations lying on the Pareto frontier.



**Figure 14. Pareto frontier obtained for case 2.**

If system cost and unavailability requirements:

- Max cost: 5 \$/hr
- Max unavailability: 2.0 E-3

are imposed to the set of points lying on the Pareto frontier, then it is possible to obtain 15 system configurations that satisfy these requirements as indicated in Table 4. Note that in these 15 system configurations, the resulting options chosen for the majority of the pumps is identical (i.e., P1, P2, P3). In this case, greater variability involves the choice of maintenance strategy for the valves which mostly ranges between options B and C.

At this point the decision maker can chose the best solution by prioritizing either system cost or unavailability. While system unavailability does not change much among the 15 system configurations, system costs has a bit higher degree of variability (i.e., 4.8 – 5 \$/hr). Given this, a possible course of action would be to choose system configuration #3: cheaper solution which satisfies unavailability constraints.

However, it should be noticed that the selection of an “optimal” solution in this case is not as straightforward as in the example presented in Section 6.1 (see Figure 13 and Table 2). In the current example there is a more gradual trade-off between system reliability and cost; thus, achieving and “optimal” decision is not as clear-cut discrimination. Hence, in this case, other considerations (e.g. availability of spare parts) would likely have a larger influence on the decision.

**Table 4. Case 2: system configurations lying in the Pareto frontier and satisfying the cost and unavailability constraints.**

#	V1	V2	V3	V4	V5	V6	V7	P1	P2	P3	Cost	Unav.
3	B	B	C	C	C	C	C	A	A	A	4.802	0.00120
4	B	B	B	C	C	C	C	A	A	A	4.8455	0.00120
5	B	B	C	B	C	C	C	A	A	A	4.8455	0.00120
6	B	B	C	C	C	B	C	A	A	A	4.8455	0.00120
7	B	B	B	B	C	C	C	A	A	A	4.889	0.00120
8	B	B	B	C	C	B	C	A	A	A	4.889	0.00120
9	B	B	C	B	B	C	C	A	A	A	4.889	0.00120
10	B	B	C	B	C	B	C	A	A	A	4.889	0.00120
11	B	B	B	B	C	B	C	A	A	A	4.9325	0.00120
12	B	B	C	B	B	B	C	A	A	A	4.9325	0.00120
13	B	B	C	B	C	B	B	A	A	A	4.9325	0.00120
14	B	B	B	B	B	B	C	A	A	A	4.976	0.00120
15	B	B	B	B	C	B	B	A	A	A	4.976	0.00120
16	B	B	C	B	B	B	B	A	A	A	4.976	0.00120
17	C	C	C	C	C	C	C	A	A	A	4.9912	0.00119

## 5.4 Maintenance Schedule Optimization

This use case focuses on the optimization of scheduling of maintenance operations for a simplified example system on the secondary side of a PWR composed of seven components (i.e., M1 through M7). Table 5 lists, for each component, the Mean Time To Repair (MTTR) and loss of power generation caused by the failure of the component while the component is under repair. It is here assumed the reactor is generating a rated power set to 1 GW.

**Table 5. Data for the maintenance optimization use case.**

Component	MTTR [h]	Power Loss
M1	10	10%
M2	12	15%
M3	6	15%
M4	9	10%
M5	10	20%
M6	7	12%
M7	12	20%

**Table 6. Schedule options for each maintenance activity.**

	T1	T2	T3	T4	T5	T6
Budget	50K	90K	90K	90K	70K	40K
M1-A	(40,0.2)					
M1-B		(40,0.25)				
M1-C			(40,0.3)			
M2-A		(50,0.1)				
M2-B			(50,0.2)			
M2-C				(50,0.35)		
M3-A			(35,0.2)			
M3-B				(35,0.2)		
M3-C						(35,0.4)
M4-A				(40,0.2)		
M4-B					(40,0.3)	
M4-C						(40,0.3)
M5-A		(45,0.1)				
M5-B			(45,0.2)			
M5-C				(45,0.3)		
M6-A	(25,0.2)					



M6-B		(25,0.3)				
M6-C			(25,0.35)			
M7-A			(30,0.3)			
M7-B				(30,0.35)		
M7-C					(30,0.35)	

Table 6 lists six time instances (i.e., T1 through T6) where it is possible to perform system maintenance. Each time instance is characterized by a maximum available budget to perform maintenance. Table 6 also provides three schedule options (i.e., A, B, or C) for each component. Each option is set on a specific time instance. In addition, each option is characterized by a tuple of values: cost and SSC failure probability  $P_{Fail}$ .

The goal is to determine the optimal maintenance schedule which minimizes plant risk. In this case, the risk  $risk_i$  for each component  $i$  is calculated as follows:

$$risk_i [GWh] = p_{fail}^i \cdot MTTR^i \cdot PowerLoss^i \quad (6)$$

Hence, the goal is to minimize the function:

$$risk [GWh] = \sum_{i=1}^7 risk_i \quad (7)$$

Table 7 expands Table 6 by including the risk  $risk_i$  associated with each component  $i$ .

By employing the Multi-Knapsack Problem (MKP) methods shown in [3] we were able to obtain the solution to this use case which is summarized in Table 8. For each component, the optimal schedule which satisfies the budget requirement and minimizes overall risk is presented.

**Table 7. Summary of the risk data for each maintenance activity.**

	T1	T2	T3	T4	T5	T6	MTTR [h]	Power Loss	Failure Probability	Risk
Budget	50K	90K	90K	90K	70K	40K				
M1-A	40						10	10%	0.2	0.2
M1-B		40					10	10%	0.25	0.25
M1-C			40				10	10%	0.3	0.3
M1-DoNothing							10	10%	1	1
M2-A		50					12	15%	0.1	0.18
M2-B			50				12	15%	0.2	0.36
M2-C				50			12	15%	0.35	0.63

M2-DoNothing							12	15%	1	1.8
M3-A			35				6	15%	0.2	0.18
M3-B				35			6	15%	0.2	0.18
M3-C						35	6	15%	0.4	0.36
M3-DoNothing							6	15%	1	0.9
M4-A				40			9	10%	0.2	0.18
M4-B					40		9	10%	0.3	0.27
M4-C						40	9	10%	0.3	0.27
M4-DoNothing							9	10%	1	0.9
M5-A		45					10	20%	0.1	0.2
M5-B			45				10	20%	0.2	0.4
M5-C				45			10	20%	0.3	0.6
M5-DoNothing							10	20%	1	2.0
M6-A	25						7	12%	0.2	0.168
M6-B		25					7	12%	0.3	0.252
M6-C			25				7	12%	0.35	0.294
M6-DoNothing							7	12%	1	0.84
M7-A			30				12	20%	0.3	0.72
M7-B				30			12	20%	0.35	0.84
M7-C					30		12	20%	0.35	0.84
M7-DoNothing							12	20%	1	2.4

**Table 8. Optimal maintenance schedule.**

	T1	T2	T3	T4	T5	T6	MTTR [h]	Power Loss	Failure Probability	Risk
Budget	50K	90K	90K	90K	70K	40K				
M1-B		40					10	10%	0.25	0.25
M2-B			50				12	15%	0.2	0.36
M3-B				35			6	15%	0.2	0.18
M4-A				40			9	10%	0.2	0.18
M5-A		45					10	20%	0.1	0.2
M6-A	25						7	12%	0.2	0.168
M7-A			30				12	20%	0.3	0.72
Total	25	85	80	75	0	0				2.058

## 6. MODEL-BASED OPTIMIZATION METHODS

Model-based optimization methods are designed for a different kind of problem compared to the data-based optimization methods. Data-based optimization aims to find the global maxima or minima of analytical functions (e.g., algebraic sum of NPV terms) that are relevant to the application at hand (e.g., optimal replacement schedule). In other terms, the data that are provided as input are limited and are assumed to be constant throughout the optimization process.

On the other hand, model-based optimization methods are designed to find the maxima or minima of a generic model (i.e., an external model) where the response of the model can change throughout the optimization process. In simple terms, data-based methods can be seen as a sub-case of model-based methods.

In mathematical form we are dealing with models that can be considered as a black box where we can define its input and output variables:

- $\mathbf{x}$ : input variable,  $\mathbf{x} = [x_1, \dots, x_N]$
- $\mathbf{y}$ : output variable,  $\mathbf{y} = [y_1, \dots, y_M]$ .

A model-based optimization method aims to minimize (or maximize) one element<sup>b</sup> of the output variables:

$$\begin{aligned}
 & \min_{\mathbf{x}} && y_1 \\
 & s. t. && \mathbf{x} \in \Xi \\
 & && \mathbf{G}(\mathbf{y}) \leq 0
 \end{aligned} \tag{8}$$

<sup>b</sup> Note that here we are focusing on single-objective optimization problem; another class of optimization problems is the multi-objective one. This class of mathematical optimization problems involves more than one objective function to be optimized simultaneously. We approached this class of optimization problems using the Pareto frontier analysis (see Section 5.1).

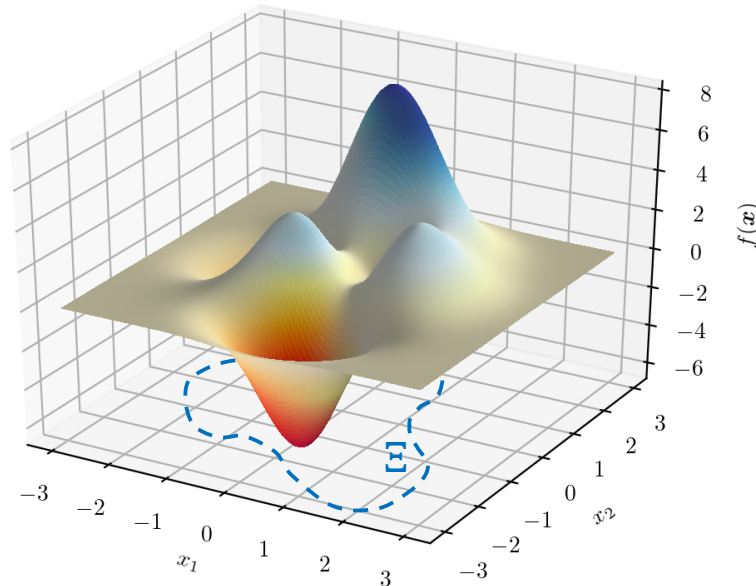
In this case, it is a model (e.g., system code, PRA model) that determines  $\mathbf{y}$  given  $\mathbf{x}$ :

$$\mathbf{y} = \mathbf{F}(\mathbf{x}) \quad (9)$$

In most of our applications, when provided  $\mathbf{x}$ ,  $\mathbf{y}$  is unique (i.e., the model does not possess any stochastic behavior). Note that two possible types of constraints can be introduced:

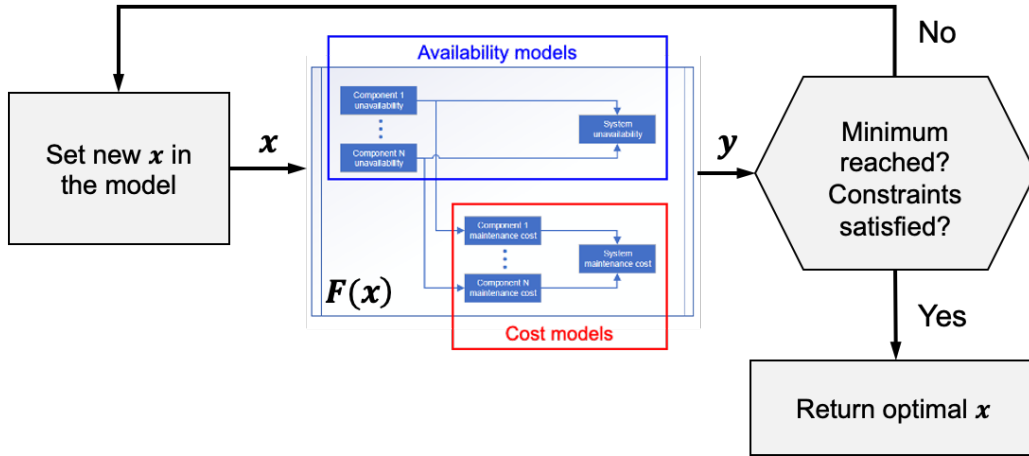
- $\mathbf{x} \in \Xi$  : explicit constraints which limit the range of variability of the input variables
- $\mathbf{G}(\mathbf{y}) < 0$  : implicit constraints which limit the range of variability of other output variables.

An example of an optimization problem is shown in Figure 15 where the objective is to find the minimum of a function  $y = f(\mathbf{x})$  where  $\mathbf{x}$  is a 2-dimensional space  $\mathbf{x} = [x_1, x_2]$ . In this case  $\Xi$  is explicitly shown in Figure 15 as a finite region in the  $[x_1, x_2]$  space.



**Figure 15. Graphical representation of the continuous optimization problem.**

While the data-based methods tend to be not computationally expensive, the model-based methods tend to require more computational resources since the model under consideration might have to be run a large number of times and each model evaluation might take a considerable amount of time (from minutes to hours) to execute.



**Figure 16. Model-based optimization scheme.**

Figure 16 shows in a flow diagram how a model-based optimization can be performed. In the RIAM application, the employed models are currently under development within the PHM project (refer to [13] for a complete overview of the developed models). In more detail, the function  $F(\mathbf{x})$  can be modeled by linking system and component reliability and cost models; afterward, it is possible to iteratively loop over the decision variable  $\mathbf{x}$  to obtain the optimal value of  $\mathbf{x}$  that minimize/maximize the objective function  $y_1$  (e.g., minimization of operating costs, maximization of system availability). As an example,  $\mathbf{x}$  can represent a candidate maintenance schedule to be performed on a system.

We have focused on the development and use of two main classes of optimization methods: continuous and discrete. Continuous methods perform optimization on a set of variables which can take on an uncountable set of values. Discrete methods act on variables defined over finite or countably infinite sets. For the scope of the RIAM project, we are concentrating on one method for each class: the gradient based method (continuous class), and genetic algorithms (discrete and, in the future, also continuous] class). The following two sections describe in more detail these two classes of methods and their application for specific RIAM problems of interest.

## 6.1 Continuous Methods

This section discusses various approaches to evaluate and optimize decisions typically encountered in operating NPPs. Since there exists a vast literature on the methods presented here the reader is referred to the literature for details. All of the techniques described in this section are described in standard operations research texts at the undergraduate and graduate level [25].

### 6.1.1 Gradient-Based Methods

The optimization problem described in Equation 8 can be numerically solved by employing gradient based optimization algorithms. Gradient based algorithms are first-order iterative optimization algorithms and they are ideal for this kind of application. The objective is to find the minimum of a function  $F(\mathbf{x})$ : starting from an initial point  $\mathbf{x}^0$ , this is performed by determining at each iteration  $r$  the gradient of  $F(\mathbf{x})$ ,  $\nabla F(\mathbf{x})$ , and moving to the next point in the direction of the gradient of the function at the current point.

From a point  $\mathbf{x}^r$  determined at iteration  $r$ , the point  $\mathbf{x}^{r+1}$  at iteration  $r + 1$  is calculated as:

$$\mathbf{x}^{r+1} = \mathbf{x}^r - \gamma \cdot \nabla F(\mathbf{x}) \quad (10)$$

The sequence:

$$(\mathbf{x}^0, F(\mathbf{x}^0)) \rightarrow (\mathbf{x}^1, F(\mathbf{x}^1)) \rightarrow (\mathbf{x}^2, F(\mathbf{x}^2)) \rightarrow \dots$$

converges to a local minima of  $F(\mathbf{x})$ .

### 6.1.2 Stochastic Methods

Stochastic methods are a variation of the one shown in Section 7.1.1. They introduce stochastic elements in the selection of point  $\mathbf{x}^{r+1}$  given  $\mathbf{x}^r$  obtained from the previous iteration.

- *Simultaneous perturbation stochastic approximation (SPSA)*: The goal of SPSA is to introduce a stochastic element in the calculation of the gradient  $\nabla F(\mathbf{x})$  in Equation 10:

$$\nabla F(\mathbf{x}) = \frac{F(\mathbf{x} + \epsilon) - F(\mathbf{x})}{\epsilon} \quad (11)$$

where  $\epsilon$  is a random perturbation vector.

- *Simulated Annealing (SA)*: SA method determines  $\mathbf{x}^{r+1}$  by first determining a candidate neighbor  $\tilde{\mathbf{x}}$  of  $\mathbf{x}^r$  (which is randomly generated) and it accepts it with a probability proportional to  $F(\tilde{\mathbf{x}}) - F(\mathbf{x}^r)$ .

## 6.2 Discrete Optimization Methods

An important class of optimization methods, especially for the RIAM project is the one that includes all the discrete optimization algorithms (see Table 9). These algorithms can be summarized as follows:

$$\begin{aligned} & \text{maximize} \quad \sum_{n=1}^N v_n x_n \\ & \text{subject to:} \quad \sum_{n=1}^N w_{n,i} x_n < W_i \\ & \quad \quad \quad x_n \geq 0 \text{ and } x_n \in \mathbb{N} \end{aligned} \quad (12)$$

Table 10 summarizes the direct application of the discrete optimization problems listed in Table 9 for specific RIAM related applications.

The following is a list of discrete optimization algorithms that can be employed to solve discrete optimization algorithms:

- Branch and bound
- Dynamic programming
- Constraint programming
- Local search
- Mixed integer programming

- Simulated annealing
- Genetic algorithms

**Table 9. Summary of discrete optimization problems.**

<b>Problem</b>	<b>Input data</b>	<b>Constraints</b>	<b>Model</b>	<b>Objective function</b>
Traveling salesman	N cities, neighbor cities (graph structure)	City can be visited once	Sum of paths	Minimize route
Postman	Graph structure	Each node can be visited more than once	Sum of paths	Minimize route
Knapsack	N objects with associated value $v$ and weight $w$	Sum of weights $< M$	Sum of weights	Maximize value
Bin packing problem	N objects with weight $w$	Bin capacity $M$	Sum of weights for each bin	Minimize number of bins
Scheduling	N jobs with associated completion time, $m$ machines	Sequential execution of jobs on each machine	Sum of completion time on each machine	Minimize completion time

**Table 10. RIAM application of discrete optimization problems.**

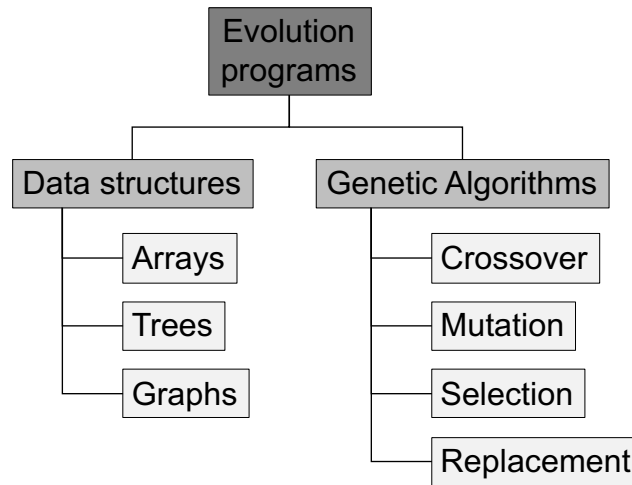
<b>Application</b>	<b>Related problem</b>
Maintenance scheduling	Scheduling, bin packing, postman, traveling salesman
Replacement scheduling	Knapsack, scheduling

### 6.2.1 Genetic Algorithms

GAs represent a relevant class of optimization methods for both continuous and discrete optimization problems. For the scope of the RIAM project we have focused on the development of GAs for discrete optimization problems. The possibility to extend the capabilities of developed GA methods to be able to deal with continuous variables is being scrutinized as well.

In addition, GAs are able to deal with any type of data structure. By data structure, we intend any form of encoding information into a digital format. While in FY-20 we have focused on arrays, GAs can act upon other data structures such as trees and graphs.

The combination of data structure and specific genetic algorithm method (e.g., crossover mutation, selection, replacement) generates what is called in the literature [5] an evolution program (see Figure 17).



**Figure 17. Data structures and methods employed in genetic algorithms that can be employed to develop evolutionary programs.**

From a high level perspective, these methods act on a population of sampled points  $(\mathbf{x}, F(\mathbf{x}))$  (rather than focusing on a one-sample-at-a-time mindset) and they iteratively combine pairs of points to generate a new generation of points with higher quality.

An initial population of  $N$  elements is initially generated (e.g., by Monte-Carlo sampling) and evaluated. Each element  $(\mathbf{x}, F(\mathbf{x}))$  of the population has the input coordinates  $\mathbf{x}$  encoded into a discrete form, a genotype, while the  $F(\mathbf{x})$  term is encoded into a fitness value  $\tilde{f}$ . The genotype form of  $\mathbf{x}$  (here indicated as  $\tilde{\mathbf{x}}$ ) is called a data structure and it can be of several forms depending on the application. In this report we focus on arrays of discrete values. More advanced data structures can be matrices, tree structures or graph structures (see Figure 17). When dealing with arrays of length  $L$  of discrete values, several options can be chosen: array of  $L$  binary values, array of  $L$  integers, combination of  $L$  integers, permutation of  $L$  integers.

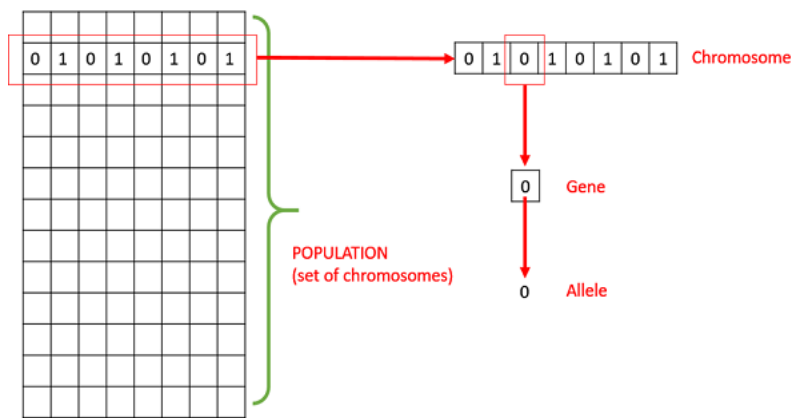
### 6.2.2 GA Data Structures

Below is a list of typical nomenclature related to evolutionary algorithms (see Figure 18):

- Phenotype space – The actual real problem solution space, comprising of solutions in the raw (non-computational) representations
- Genotype Space – The computational space comprised of all candidate solutions after encoding to a computational representation
- Population – A subset of all candidate solutions in the Genotype (encoded) space
- Chromosomes (Individual) – A single possible solution of the problem at hand taken from that population



- Gene – A single element in the chromosome.
- Allele – The value in the Gene
- Mating (Reproduction) Pool – a collection of parents used to create a new generation
- Fitness Function – The function used to rank the solutions (elitism). It might or might not be the same as the objective function
- Decoding and Encoding – are the optional processes to convert Phenotype representation (real variables) into Genotype representation (computational representation)
- Reproduction Operations – operations that alter the composition of a certain chromosome, i.e., crossover, mutation, or selection.



**Figure 18. Graphical representation of the GA data structures.**

### 6.2.3 GA Workflow

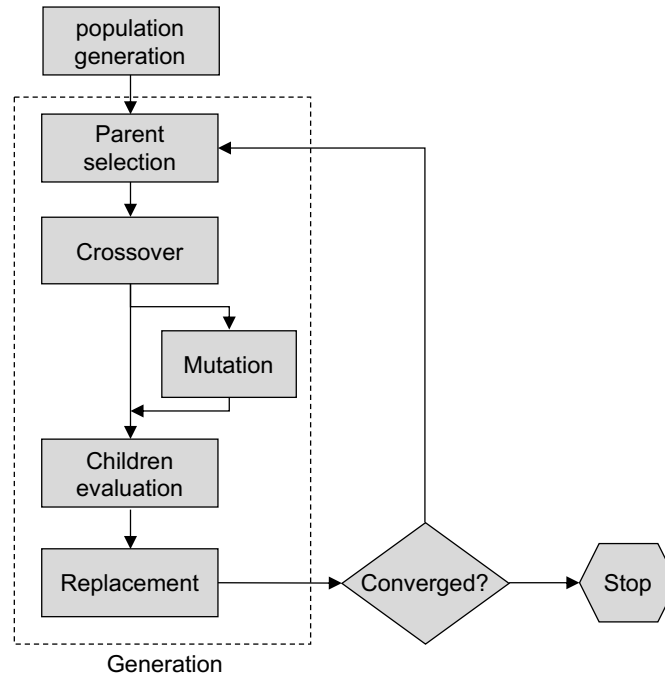
The main operators that are being employed by GAs are the following:

- Crossover: the encodings of two chromosomes are mixed to generate two new encodings
- Mutation: the encoding of a chromosome is altered by randomly changing the value of a single element of the chromosome
- Replacement: the population of chromosomes is updated by removing chromosomes with low fitness or high generational age value and keeping chromosomes with high fitness or low generational age.

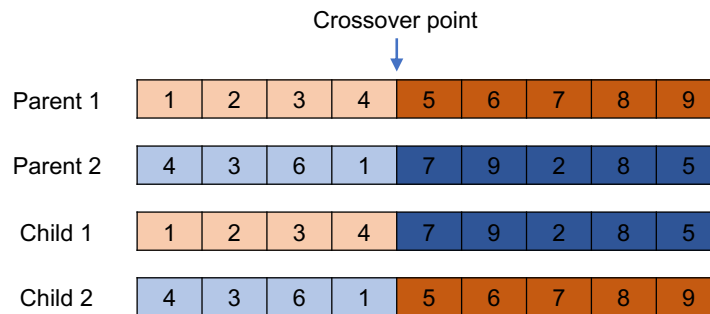
The main structure of a GA optimization algorithm is as follows (see Figure 19):

1. Create initial population: perform uniform sampling of the region of interest:
  - Monte-Carlo sampling of  $N$  samples  $(\mathbf{x}, F(\mathbf{x}))_n$ ,  $n = 1, \dots, N$
2. Perform a genotype representation according to the problem under investigation
3. Calculate fitness of each chromosome:  $(\mathbf{x}, F(\mathbf{x}))_n \rightarrow (\tilde{\mathbf{x}}, \tilde{f})_n$
4. Reproduction: create the new generation of offspring from current population:

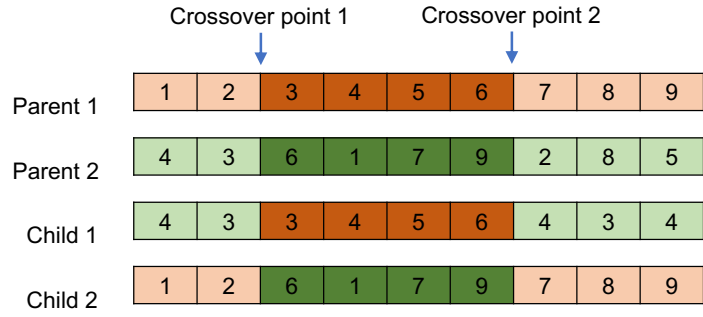
- a. Perform parent selection from the population based on their fitness
  - b. Perform crossover: creation of child population (see Figure 20 and Figure 21) by mixing chromosome structure of parents
5. Perform random mutation on the generated offspring (see Figure 22)
  6. Evaluate offspring (i.e., determine  $F(\mathbf{x})$ ) and calculate their fitness  $\tilde{f}$
  7. Return to Step 4 until convergence is met.



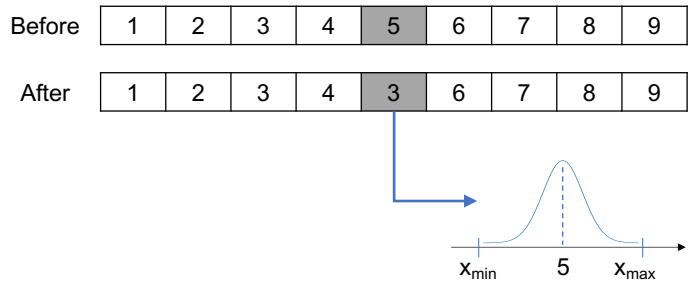
**Figure 19. GA workflow.**



**Figure 20. Single point crossover operation.**



**Figure 21. Two-points crossover operation.**



**Figure 22. Example of a single bit mutation of a chromosome.**

### 6.3 GA Development

The development of GA based methods has started in FY-20 within the RAVEN statistical framework. This development has been shared with another project within the RISA pathway: the plant reload process optimization project. A new optimizer class has been added to RAVEN which encodes the workflow shown in Figure 19 as indicated in Figure 23. As an initial step, the development has started for problems that are discrete in nature (i.e., the input space is discrete in nature). For the scope of the RIAM project, the variables can have only positive integer values.

### 6.4 RIAM Applications for GA methods

The following sections provide an overview of the specific problems that can be solved using GAs coupled with external models. Note that these represent standard problems and are described in the Operations Research literature. Application of several of these approaches to RIAM were discussed in detail in previous project report (see reference [19]) and are not repeated here.

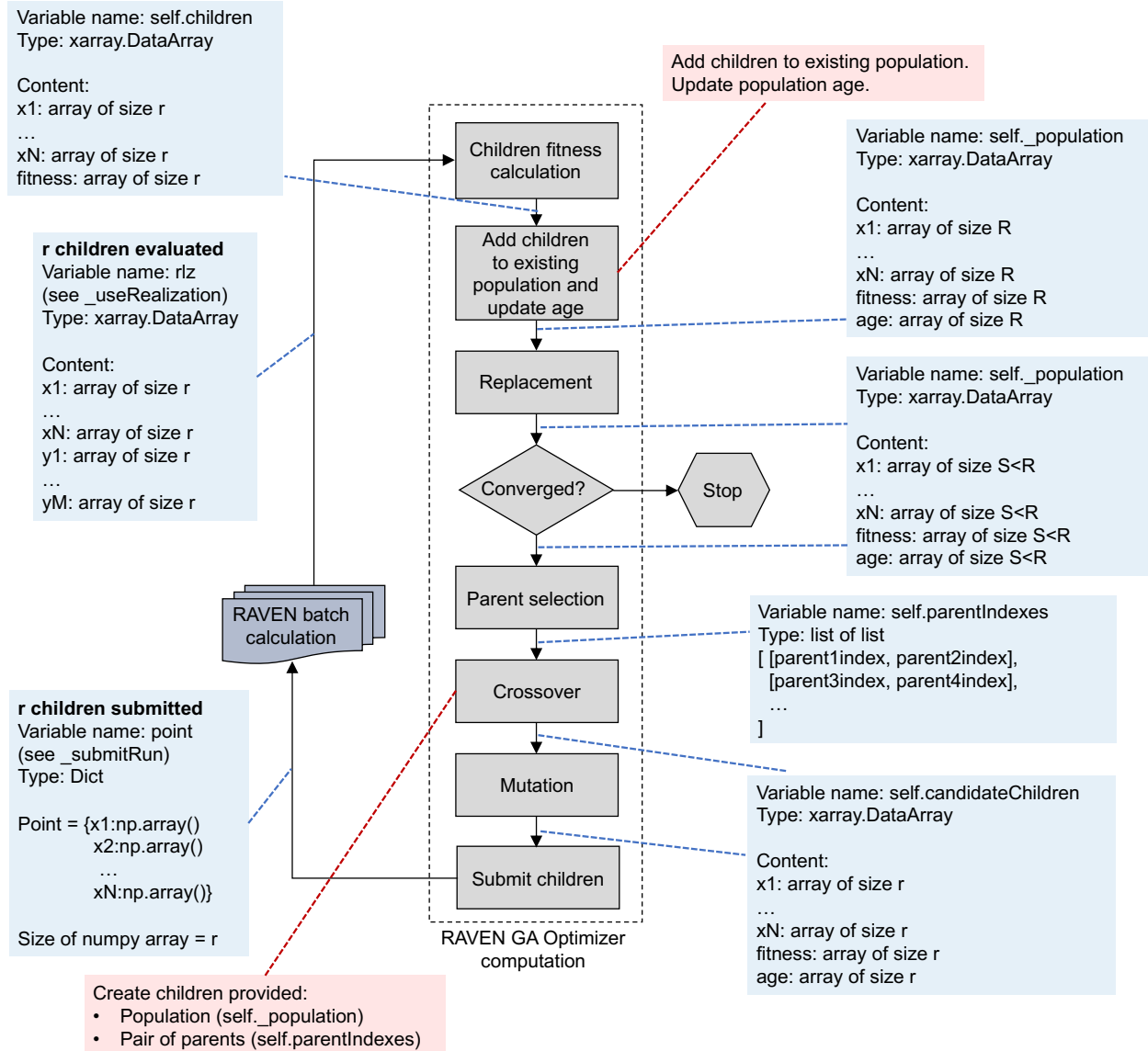
#### 6.4.1 Knapsack Problem

Given:

- A knapsack having capacity  $C$
- $N$  objects having value  $v_n$  and cost  $c_n$  ( $n = 1, \dots, N$ )

Determine  $\mathbf{x} = [x_1, \dots, x_N]$  such that:

- $x_n \in \{0,1\}$
- the function  $\sum_{n=1}^N v_n x_n$  is maximized
- $\sum_{n=1}^N c_n x_n \leq C$



**Figure 23. Data workflow for the GA methods developed in RAVEN.**

### 6.4.2 Knapsack Problem (Modified)

Given:

- A knapsack having capacity  $C$

- $N$  objects having value  $v_n$  and cost  $c_n$  ( $n = 1, \dots, N$ )

Determine  $\mathbf{x} = [x_1, \dots, x_N]$  such that:

- $x_n \in \mathbb{Z}$
- the function  $\sum_{n=1}^N v_n \delta(x_n - n)$  is maximized
- $\sum_{n=1}^N c_n \delta(x_n - n) \leq C$

where

$$\delta(x_n - n) = \begin{cases} 1 & \text{if } x_n = n \\ 0 & \text{otherwise} \end{cases}$$

is the standard Dirac delta function.

### 6.4.3 Multiple-Knapsack Problem

Given:

- $M$  knapsacks having capacity  $C_m$  ( $m = 1, \dots, M$ )
- $N$  objects having value  $v_n$  and cost  $c_{n,m}$  ( $n = 1, \dots, N$  and  $m = 1, \dots, M$ )

Determine  $x_{n,m}$  such that:

- $x_{n,m} \in \{0,1\}$  where
- $$x_{n,m} = \begin{cases} 1 & \text{if object } n \text{ is assigned to knapsack } m \\ 0 & \text{otherwise} \end{cases}$$
- the function  $\sum_{n=1}^N v_n x_n$  is maximized
  - $\sum_{n=1}^N c_{n,m} x_n \leq C_m$

### 6.4.4 Multiple-Knapsack Problem (Modified)

Given:

- $M$  knapsacks having capacity  $C_m$  ( $m = 1, \dots, M$ )
- $N$  objects having value  $v_{n,m}$  and cost  $c_{n,m}$  ( $n = 1, \dots, N$ )

Determine  $\mathbf{x} = [x_1, \dots, x_N]$  such that:

- $x_n \in [0, \dots, M]$ , e.g.,  $x_1 = 4$  means that object 1 is selected for window 4
- the function  $\sum_{n=1}^N \sum_{m=1}^M v_{n,m} \delta(x_n - m)$  is maximized
- $\sum_{n=1}^N \sum_{m=1}^M c_{n,m} \delta(x_n - m) \leq C_m$

where

$$\delta(x_n - m) = \begin{cases} 1 & \text{if } x_n = m \\ 0 & \text{otherwise} \end{cases}$$

### 6.4.5 Traveling Salesman Problem

Given:

- $N$  cities  $l_n$  ( $n = 1, \dots, N$ )
- Connection matrix  $C = [c_{i,j}]$  where  $c_{i,j} = \begin{cases} 1 & \text{a path exist from city } i \text{ to city } j \\ \text{inf} & \text{otherwise} \end{cases}$
- Distance matrix  $D = [d_{i,j}]$  where  $d_{i,j}$  is the distance between city  $i$  and city  $j$

Determine a path  $x = [x_1, \dots, x_N]$  such that:

- $x_n \in [0, \dots, N]$ , e.g.,  $x_2 = 4$  means that city  $l_4$  has been the second one to be visited
- the function  $\sum_{n=2}^N d_{n-1,n} x_{n-1,n}$  is minimized; here,  $d_{n-1,n}$  represents the distance between the city visited at step  $n$  and the one visited at step  $n - 1$

## 6.5 Initial Testing of GA Methods

An initial testing phase has started during FY-20 with the objective of assessing GA performances and validate their results against other methods. As an example, we tested GA methods vs. LOGOS models for a simple multiple knapsack problem specified in Table 11. Note that this example does not provide direct application of any NPP asset management use cases, but it is provided here for validation and testing purposes.

Given a set of ten projects, an estimated NPV and its cost are provided in input for each project. Given a five-year planning scenario, the goal is to choose the optimal set of projects and their actuation year provided the budget constraints indicated in Table 12 which maximizes the overall NPV. Table 12 shows a deterministic point forecast for the available capital budget for the next five years for our example problem.

By employing the multiple knapsack problem available in LOGOS (see Section 7) it was possible to obtain an overall NPV equal to \$20M which was obtained by choosing the combination of projects indicated in Table 13.

When approaching this problem using the GA methods, we have developed a model which computes overall NPV provided the candidate project schedule. This model penalizes those choices that exceed the budget constraints indicated in Table 12. Note that this is still a linear problem, however this model can be customized to solve more complex cases that cannot be directly solved using the available LOGOS models (e.g., correlations among NPV projects).

We employed the GA optimization method summarized in Figure 24 where we specified:

- A random selection of the parents weighed by their fitness
- A single point crossover method (see Figure 20)
- A single bit mutation method (see Figure 22)

We let the optimization method operate for 300 generations and were able to obtain a family of optimal schedules (including the one listed in Table 13) with an overall NPV equal to \$20M (as predicted by the LOGOS model).

**Table 11. Multiple knapsack problem employed to test GA performances.**

Project	NPV [M\$]	Cost [K\$]
1	1	300
2	3	200
3	4	500
4	2	300
5	1	200
6	2	400
7	3	600
8	4	300
9	2	500
10	4	300

**Table 12. Point estimates of the annual capital budget over a five-year planning horizon for the multiple knapsack problem test case.**

Capital budget [K\$]				
Y1	Y2	Y3	Y4	Y5
400	500	400	500	500

**Table 13. LOGOS solution for the multiple knapsack problem test case indicated in Table 11 and Table 12.**

Project	Year
1	-
2	Y5
3	Y2
4	Y3
5	Y4
6	Y1
7	-
8	Y4
9	-
10	Y5

```

<GParams>
  <populationSize>60</populationSize>
  <parentSelection>rouletteWheel</parentSelection>
  <reproduction nParents="4">
    <crossover type="onePointCrossover">
      <points>random</points>
      <crossoverProb>0.9</crossoverProb>
    </crossover>
    <mutation type="swapMutator">
      <locs>random</locs>
      <mutationProb>0.1</mutationProb>
    </mutation>
  </reproduction>
  <fitness type="logistic">
    <a>0.2</a>
    <b>13.0</b>
  </fitness>
  <survivorSelection>fitnessBased</survivorSelection>
</GParams>

```

Figure 24. GA workflow specified in RAVEN for the multiple knapsack problem test case.

## 7. METHODS DEVELOPMENT: LOGOS

LOGOS is a software package which contains a set of discrete optimization models that can be employed for capital budgeting optimization problems. More specifically, provided a set of items (characterized by cost and reward values) and constraints, these models select the best combination of items which maximize overall reward and satisfies the provided constraints.

The developed models are based on different versions of the knapsack optimization algorithms. Two main classes of optimization models have been initially developed: deterministic and stochastic. Stochastic optimization models evolve deterministic models by explicitly considering data uncertainties (associated with constraints or item cost and reward). These models can be employed as stand-alone models or interfaced with the INL developed RAVEN code to propagate data uncertainties and analyze the generated data (i.e., sensitivity analysis).

One of LOGOS' objectives is to optimize capital budgeting and SSC replacements to support risk-informed decisions in NPP operations. Since capital budgeting is influenced by various factors such as markets, safety, and regulations, its decision-making process should consider the relevant factors for balancing risks, costs, and profits. In the last year, LOGOS was developed to provide both risk-free approaches based on deterministic optimization schemes and risk-informed approaches based on stochastic optimization schemes for NPP asset management. In the literature, the problems of capital budgeting and SSC replacement optimization can be represented by the variants of the knapsack problem. In the last year, LOGOS was developed to handle several representative variants that are generally adopted for the capital budgeting analysis:

- Unbounded and bounded single knapsack problem (SKP): single budget constraint
- Multidimensional knapsack problem (DKP): multiple budget constraints, i.e. capital funds, operation and maintenance funds, and time-dependent funds and multi-year funds
- Multiple knapsack problem (MKP): budget constraints for maintenance of multi-units of NPPs in parallel



- Multiple-choice knapsack problem (MCKP): multiple ways to carry out each investment/replacement
- Multidimensional multiple-choice knapsack problem (MMKP): considering both multiple ways to carry out each investment/replacement and multiple budget constraints mentioned in the DKP method.

One limitation of the deterministic optimization models for capital budgeting is that they do not account for risk or uncertainty in profit and cost streams associated with individual projects, nor do they account for risk in future resource availability. LOGOS was extended to handle the risks and uncertainties during the asset management analysis via a two-stage stochastic optimization scheme to provide priority lists to decision-makers in support of risk-informed decisions.

These methods are being developed and maintained in an INL GitLab repository as shown in Figure 25.

**LOGOS** Project ID: 1740

229 Commits 12 Branches 0 Tags 13 MB Files 103.6 MB Storage

Operation Optimization Toolkit

master LOGOS / +

History Find file Web IDE Clone

Merge branch 'wancg/update\_TEAL' into 'master'   
 Mandelli, Diego authored 3 weeks ago

README CI/CD configuration Add LICENSE Add CHANGELOG Add CONTRIBUTING

Add Kubernetes cluster

Name	Last commit	Last update
.gitlab	update gitlab CI and issue template	4 months ago
doc	Adding more directory fixes.	2 months ago
pyomo_models	address comments from reviewer	4 months ago
raven_models/NPV_models	add raven external models for NPV calculations	10 months ago
scripts	add test description scripts	9 months ago
src	update changes with TEAL	3 weeks ago
tests	change regulatorymandated to mandatory	2 months ago
.gitignore	add module CapitalInvestments	1 year ago
.gitlab-ci.yml	update ci to reflect TEAL	3 weeks ago
README.md	Update README.md	4 months ago
__init__.py	change Logos to LOGOS	5 months ago
build.sh	restructure Capital Investments Optimizations	9 months ago
dependencies.xml	convert dependencies to use xml format	1 month ago
logos	some update	9 months ago
run_tests.py	update run_tests	6 months ago
setup.py	restructure Capital Investments Optimizations	9 months ago

**README.md**

### Logos: Operation Optimization Toolkit

LOGOS is a software package which contains a set of discrete optimization models that can be employed for capital budgeting optimization problems. More specifically, provided a set of items (characterized by cost and reward values) and constraints, these models select the best combination of items which maximizes overall reward and satisfies the provided constraints. The developed models are based on different versions of the knapsack optimization algorithms. Two main classes of optimization models have been initially developed: deterministic and stochastic. Stochastic optimization models evolve deterministic models by explicitly considering data uncertainties (associated to constraints or item cost and reward). These models can be employed as stand-alone models or interfaced with the INL developed RAVEN code to propagate data uncertainties and analyze the generated data (i.e., sensitivity analysis).

**Figure 25. Snapshot of the LOGOS repository.**

In FY-20, LOGOS capabilities were further extended to provide more robust RIAM by adding both Conditional Value at Risk (CVaR) optimization and Distributionally Robust Optimization (DRO). CVaR, also called mean excess loss, mean shortfall, or tail (VaR), is defined as the conditional expected loss under the condition that it exceeds VaR. For the value-at-risk capital budgeting problem, LOGOS tries to maximize a reward functions (e.g., expected NPVs) while minimizing the risk, as measured by CVaR. The previously developed stochastic optimization approach assumes that the decision maker has complete knowledge about the underlying uncertainty through a known probability density function (PDF). The PDF of the random parameters is inferred from historical data, prior beliefs, or expert opinions. In DRO, on the other hand, the designer can provide a set of possible pdfs and DRO then optimize for the worst-case pdf within this set.

In the following sections, we provide the mathematical representations of CVaR and DRO optimizations for different situations of capital asset management. The notation and formulation of the CVaR and DRO optimizations are as follows:

*Indices and Sets:*

$i, i' \in N$	candidate projects
$j \in J_i$	options for selecting project $i$ , e.g., initiate project $i$ in year $t$ or $t + 2$ and in a standard (three year) or in an expedited (two year) manner. Note that the last option for project $i$ is always used to indicate “non-selection”, i.e. the investment $i$ is not selected.
$d \in D$	types of resources, e.g., capital funds, O&M funds, labor-hours, time during outage
$t \in T$	time periods (years)
$\omega \in \Omega$	Scenarios
$\sigma \in \Omega$	Scenarios

*Data:*

$p_i^\omega$	profit of investment $i$ under scenario $\omega$ (NPV)
$p_{i,j}^\omega$	profit of investment $i$ via option $j$ under scenario $\omega$ (NPV)
$c$	available budget under scenario $\omega$
$c_d^\omega$	available budget for a resource of type $d$ under scenario $\omega$
$c_m^\omega$	available budget for unit $m$ under scenario $\omega$
$c_t^\omega$	available budget in year $t$ under scenario $\omega$
$c_{d,t}^\omega$	available budget for a resource of type $d$ in year $t$ under scenario $\omega$
$w_i^\omega$	cost of investment $i$ under scenario $\omega$
$w_{i,d}^\omega$	consumption of resource of type $d$ if investment $i$ is selected under scenario $\omega$
$w_{i,j,t}^\omega$	consumption of resource in year $t$ if investment $i$ is performed via option $j$ under scenario
$w_{i,j,d,t}^\omega$	consumption of resource of type $d$ in year $t$ if investment $i$ is performed via option $j$ under scenario $\omega$
$q^\omega$	probability of scenario $\omega$
$\lambda \in [0, 1]$	weight for CVaR analysis
$\alpha$	confidence level
$\varepsilon$	ambiguity radius for DRO analysis

$d_{\sigma,\omega}$  the Euclidean distance between scenario  $\sigma$  and scenario  $\omega$

*Decision variables:*

$$y_{i,i'} = \begin{cases} 1 & \text{if project } i \text{ has no lower priority than project } i' \\ 0 & \text{otherwise} \end{cases}$$

$$x_i^\omega = \begin{cases} 1 & \text{if project } i \text{ is selected under scenario } \omega \\ 0 & \text{otherwise} \end{cases}$$

$$x_{i,m}^\omega = \begin{cases} 1 & \text{if project } i \text{ is selected for unit } m \text{ under scenario } \omega \\ 0 & \text{otherwise} \end{cases}$$

$$x_{i,j}^\omega = \begin{cases} 1 & \text{if project } i \text{ is performed via option } j \text{ is selected under scenario } \omega \\ 0 & \text{otherwise} \end{cases}$$

$u$  variable to indicate VaR

$v^\omega$  dummy variable to linearize the CVaR analysis

$\gamma$  dummy variable for DRO analysis

$v^\sigma$  dummy variable for DRO analysis

## 8. STOCHASTIC OPTIMIZATION WITH CVAR

### 8.1 Deterministic Capital Budgeting

Consider a capital budgeting problem faced by an NPP. To begin, and so that we can better motivate what follows, we assume for the moment that we can make plans using deterministic forecasts for available annual capital budgets as well as the NPV and liability streams (i.e., capital costs) associated with selecting each project. We assume that there are some “must do” projects (e.g. those being performed to meet a regulatory requirement) and other optional projects. For the specific problem instance with which we begin, our objective is to select a set of projects in such a way that the overall NPV is maximized while respecting available capital budgets over a five-year horizon. Table 14 provides a brief description of the instance and shows the deterministic forecasts for the NPV and liability streams, for a set of 16 candidate projects.

Table 15 shows a deterministic point forecast for the available capital budget for the next five years for our example problem.

There are 6 must-do projects and 10 optional projects in this set. Hence, there are  $2^{10} = 1,024$  possible portfolios of projects. (If all 16 projects were optional, we would have  $2^{16} = 65,536$  possible portfolios.) Some portfolios are infeasible because they consume more than the available budget in one or more years. Given the small size of this problem instance, we can enumerate all feasible projects and select the one with the largest NPV. Or, we can formulate a knapsack-style optimization model – a type of integer programming problem – and solve it with open-source or commercially available software. Table 16 shows an optimal portfolio with overall NPV of \$19.90M in which we select 4 of the 10 optional projects.

**Table 14. Candidate set of 16 projects for an example problem in capital budgeting. The “Category” column distinguishes optional and must-do projects. The “NPV” column provides a point estimate of each project’s net present value. “Capital costs” provide point estimates of the liability streams induced by selecting each project over a five-year horizon. Values are in millions of dollars.**

Project	Project name	Category	Capital costs (\$M)					NPV (\$M)
			Y1	Y2	Y3	Y4	Y5	
1	HP feedwater heater upgrade	Optional	13.01	1.31	0	0	0	27.98
2	Pressurizer replacement	Must Do	9.19	0.93	0	0	0	-10.07
3	Improvement to emergency diesel generators	Optional	0	0	0	10.09	1.11	20.23
4	Secondary system PHM system	Optional	0	4.51	0.31	0.21	0	35.00
5	Replacement of two reactor coolant pumps	Must Do	0	18.63	0	0	0	-18.60
6	Seismic modification, requalification, reinforcement, improvement	Optional	0	2.44	0	0	0	9.48
7	Fire protection	Must Do	1.32	0.14	0	0	0	-1.44
8	Service water system upgrade	Optional	2.35	0	0	0	0	5.18
9	Batteries replacement	Optional	0.29	0	0	0	0	2.10
10	Replace CCW piping, heat exchangers, valves	Must Do	0	0	4.58	0.47	0	-5.03
11	Reactor vessel internals	Optional	0	19.85	0	0	0	41.14
12	Reactor vessel upgrade (head included)	Must Do	5.27	0	0	0	0	-5.25
13	Replace LP turbine	Optional	0	0	18.79	0	0	167.94
14	Replace instrumentation and control cables	Must Do	5.94	0.61	0	0	0	-6.52
15	Condenser retubing	Optional	5.26	0	0	0	0	16.72
16	Replace moisture separator reheater	Optional	3.17	0	0	0	0	8.26

**Table 15. Point estimates of the annual capital budget over a five-year planning horizon. Values are in millions of dollars.**

Capital budget [\$M]				
Y1	Y2	Y3	Y4	Y5
22.664	36.896	20.636	23.784	22.814

**Table 16. Optimal solution to the deterministic capital budgeting problem in which we select 4 of 10 optional projects, and all must-do projects. This solution respects the annual capital budgets of Table 12 and maximizes NPV, achieving a portfolio NPV of \$19.90M.**

ID	Category	Project name	Decision
1	Optional	HP feedwater heater upgrade	Do Nothing
2	Must do	Pressurizer replacement	Select
3	Optional	Improvement to emergency diesel generators	Select
4	Optional	Secondary system PHM system	Select
5	Must do	Replacement of two reactor coolant pumps	Select
6	Optional	Seismic modification, requalification, reinforcement, improvement	Select
7	Must do	Fire protection	Select
8	Optional	Service water system upgrade	Do Nothing
9	Optional	Batteries replacement	Select
10	Must do	Replace CCW piping, heat exchangers, valves	Select
11	Optional	Reactor vessel internals	Do Nothing
12	Must do	Reactor vessel upgrade (head included)	Select
13	Optional	Replace LP turbine	Do Nothing
14	Must do	Replace instrumentation and control cables	Select
15	Optional	Condenser retubing	Do Nothing
16	Optional	Replace moisture separator reheater	Do Nothing

### 8.1.1 Stochastic Capital Budgeting with Options

We now introduce three significant changes to the simple problem just sketched. First, we introduce multiple options for some of the projects. As shown in Table 17, Projects 9, 12, and 14 simply replicate the information in Table 14, labeling the only available option as “Plan A.” The other 13 projects either have two implementation options (Plans A and B) or three implementation options (Plans A, B, and C). Here, Plans B and C involve the possibility of shifting the timing of implementing the project relative to that of the nominal Plan A from Table 14.

For the optional projects, delaying the project’s start date means that we delay its benefit and hence NPV is reduced as the table shows. Delaying a must-do project delays the liability stream and increases a negative NPV, although some must-do projects cannot be delayed. (Note that in our example we assume that all of the must do projects represent regulatory requirements and provide no additional economic value to the plant other than permitting it to continue to operate.)

This increased flexibility, due to the possibility of shifting the timing, allows for greater total NPV of the selected portfolio because we can choose additional, or different and more profitable, projects while still respecting the annual capital budgets; see [19] for further details. Table 17 shows that there are three

possibilities for the optional Project 1: do nothing, Plan A, or Plan B, with plan B delaying implementation by one year, along with lower NPV. The must-do Project 2 also has three options: Plan A, B, or C. Thus, there are 9 ways to select these two projects. With this type of logic, over the 16 candidate projects that we consider, we now have over 2.5 million possible portfolios.

**Table 17. The table replicates information from Table 14 except that most of the projects now have two or three implementation options via Plan A, B, and/or C. Values are in millions of dollars.**

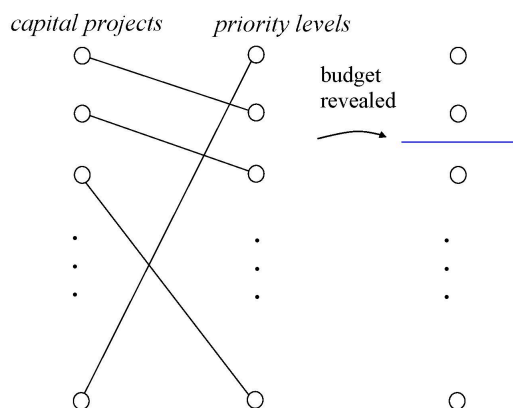
ID	Project name	Category	Options	Capital costs (\$M)					NPV (\$M)	
				Y1	Y2	Y3	Y4	Y5		
1	HP feedwater heater upgrade	Optional	Plan A, B	A:	13.01	1.31	0.00	0.00	0.00	27.98
				B:	0.00	13.01	1.31	0.00	0.00	27.17
2	Pressurizer replacement	Must do	Plan A, B, C	A:	9.19	0.93	0.00	0.00	0.00	-10.07
				B:	0.00	9.19	0.93	0.00	0.00	-9.78
				C:	0.00	0.00	0.00	9.19	0.93	-9.22
3	Improvement to emergency diesel generators	Optional	Plan A, B	A:	0.00	0.00	0.00	10.09	1.11	20.23
				B:	0.00	0.00	10.09	1.11	0.00	20.84
4	Secondary system PHM system	Optional	Plan A, B	A:	0.00	4.51	0.31	0.21	0.00	35.00
				B:	0.00	0.00	4.51	0.31	0.21	33.98
5	Replacement of two reactor coolant pumps	Must do	Plan A, B	A:	0.00	18.63	0.00	0.00	0.00	-18.60
				B:	0.00	0.00	0.00	0.00	18.63	-17.02
6	Seismic modification, requalification, reinforcement, improvement	Optional	Plan A, B, C	A:	0.00	2.44	0.00	0.00	0.00	9.48
				B:	0.00	0.00	0.00	2.44	0.00	8.94
				C:	0.00	0.00	0.00	0.00	2.44	8.68
7	Fire protection	Must do	Plan A, B	A:	1.32	0.14	0.00	0.00	0.00	-1.44
				B:	0.00	0.00	0.00	1.32	0.14	-1.32
8	Service water system upgrade	Optional	Plan A, B	A:	2.35	0.00	0.00	0.00	0.00	5.18
				B:	0.00	0.00	2.35	0.00	0.00	4.88
9	Batteries replacement	Optional	Plan A	A:	0.29	0.00	0.00	0.00	0.00	2.10
10	Replace CCW piping, heat exchangers, valves	Must do	Plan A, B, C	A:	0.00	0.00	4.58	0.47	0.00	-5.03
				B:	0.00	4.58	0.47	0.00	0.00	-5.18
				C:	0.00	0.00	0.00	4.58	0.47	-4.88
11	Reactor vessel internals	Optional	Plan A, B	A:	0.00	19.85	0.00	0.00	0.00	41.14
				B:	0.00	0.00	0.00	0.00	19.85	37.65
12	Reactor vessel upgrade (head included)	Must do	Plan A	A:	5.27	0.00	0.00	0.00	0.00	-5.25
13	Replace LP turbine	Optional	Plan A, B	A:	0.00	0.00	18.79	0.00	0.00	167.94
				B:	0.00	0.00	0.00	18.79	0.00	163.05
14	Replace instrumentation and control cables	Must do	Plan A	A:	5.94	0.61	0.00	0.00	0.00	-6.52
15	Condenser retubing	Optional	Plan A, B	A:	5.26	0.00	0.00	0.00	0.00	16.72
				B:	0.00	0.00	5.26	0.00	0.00	15.76
16	Replace moisture separator reheater	Optional	Plan A, B, C	A:	3.17	0.00	0.00	0.00	0.00	8.26
				B:	0.00	0.00	0.00	3.17	0.00	7.56
				C:	0.00	0.00	0.00	0.00	3.17	7.34

The second significant change that we introduce is to now model stochasticity in capital budgets. That is, we retain deterministic point forecasts for the capital costs and the NPV associated with each of the 16 projects. However, we assume that we have uncertain capital budgets with a total of 10 scenarios. In particular, we replace the deterministic point forecast of the annual budget in each of the five years with a discrete distribution between \$22M and \$40M as shown in Table 18.

**Table 18. We replace the point forecast of the annual capital budget over a five-year horizon in Table 12 with 10 scenarios. The probability distribution puts equal mass (probability 0.1) on each of the 10 values, and the values are perfectly correlated across time; e.g., if the budget realization is scenario 6 (S6) then the budget is \$31.0M in each of the five years. Values are in millions of dollars.**

Annual capital budget (\$M)									
S1	S2	S3	S4	S5	S6	S7	S8	S9	S10
22.0	23.8	25.6	27.4	29.2	31.0	32.8	34.6	36.4	40.0

An optimal solution to the deterministic capital budget problem involves “selecting” all of the must-do projects along with selecting a subset of the optional projects so that the annual capital budget constraint is satisfied in each of the five years. What is a solution to the analogous problem when we have uncertain capital budgets? If we could wait to see the realization of the budget prior to selecting the portfolio of projects then we would simply solve 10 separate knapsack problems, i.e., one for each budget scenario. However, if we must commit to a capital budgeting plan *prior* to realizing the budget then we can employ a priority list as depicted in Figure 26.



**Figure 26. Candidate projects (e.g., 16 projects) are listed in the left-hand column. In a priority list these are mapped to priority levels in the middle column. Then the uncertain budget is revealed. For the budget realized in the figure, the two projects with the highest priority levels can be implemented.**

An example solution under 10 budget scenarios is shown Table 19. Of course, the must-do projects are selected in all budget scenarios, as indicated by the values of “1” under scenarios S1-S10. Project 3 is also selected under all budget scenarios. The right-most column of the table indicates these seven projects are tied for priority levels 1-7. Projects 9 and 13 enter under budget scenario 2 (S2) and are at priority levels 8-9, Project 4 enters under S3 (priority 10), Projects 6 and 15 under S4 (priority 11-12), Project 16 under S6

(priority 13), and Project 8 under S7 (priority 14). Projects 1 and 11 are not selected under any budget scenario and have the lowest priority level, tied for 15-16. For simplicity of presentation the model assumes that Plan A is the only option. Note that the prioritized nature of the solution means that as we read the table from left-to-right, once a project is selected (indicated by a “1”) it must remain selected under all higher budget scenarios. The expected NPV of the prioritized solution is \$178.90M.

**Table 19. Example solution under 10 budget scenarios, S1-S10. If a project has a “0” in the corresponding entry, that project is not selected under that scenario, and it is selected under entry “1.” The must-do projects are selected under all scenarios. The right-most column indicates the priority level associated with each project. For example, there are seven projects tied for the highest priority because they are selected under all scenarios, and there are two projects tied for priority 15-16 because they are not selected under any scenario. For simplicity of presentation, this example has no timing options.**

ID	Category	Project name	S1	S2	S3	S4	S5	S6	S7	S8	S9	S10	Priority
1	Optional	HP feedwater heater upgrade	0	0	0	0	0	0	0	0	0	0	15-16
2	Must do	Pressurizer replacement	1	1	1	1	1	1	1	1	1	1	1-7
3	Optional	Improvement to emergency diesel generators	1	1	1	1	1	1	1	1	1	1	1-7
4	Optional	Secondary system PHM system	0	0	1	1	1	1	1	1	1	1	10
5	Must do	Replacement of two reactor coolant pumps	1	1	1	1	1	1	1	1	1	1	1-7
6	Optional	Seismic modification, requalification, reinforcement, improvement	0	0	0	1	1	1	1	1	1	1	11-12
7	Must do	Fire protection	1	1	1	1	1	1	1	1	1	1	1-7
8	Optional	Service water system upgrade	0	0	0	0	0	0	1	1	1	1	14
9	Optional	Batteries replacement	0	1	1	1	1	1	1	1	1	1	8-9
10	Must do	Replace CCW piping, heat exchangers, valves	1	1	1	1	1	1	1	1	1	1	1-7
11	Optional	Reactor vessel internals	0	0	0	0	0	0	0	0	0	0	15-16
12	Must do	Reactor vessel upgrade (head included)	1	1	1	1	1	1	1	1	1	1	1-7
13	Optional	Replace LP turbine	0	1	1	1	1	1	1	1	1	1	8-9
14	Must do	Replace instrumentation and control cables	1	1	1	1	1	1	1	1	1	1	1-7
15	Optional	Condenser retubing	0	0	0	1	1	1	1	1	1	1	11-12
16	Optional	Replace moisture separator reheater	0	0	0	0	0	1	1	1	1	1	13



**Table 20. The table replicates information from Table 14, and also shows the uncertain NPVs. The NPV values for the low-risk and medium-risk projects are shown for the three possible scenarios: pessimistic, most likely, and optimistic. The probabilities associated with the scenarios are given in Table 18. No-risk projects 2, 4, 5, 7, 9, 10, 12, and 14 are not subject NPV uncertainty and their values repeat those shown in Table 14.**

ID	Project name	Risk Level	Options	Capital costs (\$M)					NPV (\$M)		
				Y1	Y2	Y3	Y4	Y5	Pessimistic	Most likely	Optimistic
1	HP feedwater heater upgrade	Medium	A:	13.01	1.31	0.00	0.00	0.00	12.35	20.68	24.68
			B:	0.00	13.01	1.31	0.00	0.00	11.99	20.08	23.97
2	Pressurizer replacement	No	A:	9.19	0.93	0.00	0.00	0.00		-10.07	
			B:	0.00	9.19	0.93	0.00	0.00		-9.78	
			C:	0.00	0.00	0.00	9.19	0.93		-9.22	
3	Improvement to emergency diesel generators	Low	A:	0.00	0.00	0.00	10.09	1.11	-3.15	1.81	5.32
			B:	0.00	0.00	10.09	1.11	0.00	-3.23	1.86	5.48
4	Secondary system PHM system	No	A:	0.00	4.51	0.31	0.21	0.00		35.00	
			B:	0.00	0.00	4.51	0.31	0.21		33.98	
5	Replacement of two reactor coolant pumps	No	A:	0.00	18.63	0.00	0.00	0.00		-18.60	
			B:	0.00	0.00	0.00	0.00	18.63		-17.02	
6	Seismic modification, requalification, reinforcement, improvement	Low	A:	0.00	2.44	0.00	0.00	0.00	8.27	10.92	11.42
			B:	0.00	0.00	0.00	2.44	0.00	7.79	10.29	10.76
			C:	0.00	0.00	0.00	0.00	2.44	7.56	9.99	10.45
7	Fire protection	No	A:	1.32	0.14	0.00	0.00	0.00		-1.44	
			B:	0.00	0.00	0.00	1.32	0.14		-1.32	
8	Service water system upgrade	Medium	A:	2.35	0.00	0.00	0.00	0.00	-5.0	5.83	8.31
			B:	0.00	0.00	2.35	0.00	0.00	-4.71	5.50	7.84
9	Batteries replacement	No	A:	0.29	0.00	0.00	0.00	0.00		2.10	
10	Replace CCW piping, heat exchangers, valves	No	A:	0.00	0.00	4.58	0.47	0.00		-5.03	
			B:	0.00	4.58	0.47	0.00	0.00		-5.18	
			C:	0.00	0.00	0.00	4.58	0.47		-4.88	
11	Reactor vessel internals	Medium	A:	0.00	19.85	0.00	0.00	0.00	-10.38	16.99	30.00
			B:	0.00	0.00	0.00	0.00	19.85	-9.51	15.55	27.45
12	Reactor vessel upgrade (head included)	No	A:	5.27	0.00	0.00	0.00	0.00		-5.25	
13	Replace LP turbine	Medium	A:	0.00	0.00	18.79	0.00	0.00	119.01	133.62	139.86
			B:	0.00	0.00	0.00	18.79	0.00	115.55	129.72	135.78
14	Replace instrumentation and control cables	No	A:	5.94	0.61	0.00	0.00	0.00		6.52	-
15	Condenser retubing	Low	A:	5.26	0.00	0.00	0.00	0.00	0.28	10.63	15.05
			B:	0.00	0.00	5.26	0.00	0.00	0.27	10.02	14.19
16	Replace moisture separator reheater	Medium	A:	3.17	0.00	0.00	0.00	0.00	-0.46	4.60	6.76
			B:	0.00	0.00	0.00	3.17	0.00	-0.42	4.21	6.19
			C:	0.00	0.00	0.00	0.00	3.17	-0.37	3.74	5.50

When considering all possible plans, a different portfolio can be selected under each scenario, and so the prioritization scheme must consider more than  $(2.5 \times 10^6)^{10} \approx 1 \times 10^{64}$  solutions, although many are eliminated by feasibility with respect to the budget and others due to prioritization combinatorics. We solve this problem using an integer programming formulation, which we detail in Table 19.

The third significant change we make to our base model is that we allow for uncertainty in the NPV of the projects. In particular, we create three groups of projects, no-risk, low-risk, and medium-risk projects. No-risk projects have the NPV (and cost) values already shown in Table 17. Low-risk projects have three possible scenarios for their NPVs: a pessimistic value, most likely value, and optimistic value. All low-risk projects move in concert with their realizations, e.g., if the low-risk set consists of Projects 3, 6, and 15, if one takes its optimistic value then so do the other two. Medium-risk projects again have three realizations, are perfectly correlated within their group, and are independent of the low-risk realizations. The realizations of the low- and medium-risk projects are independent of the budget realizations leading to 90 total scenarios. Table 20 shows the NPV values for the low-risk and medium-risk projects for the three possible scenarios: pessimistic (low), most likely (medium), and optimistic (high).

**Table 21. The table shows no-risk, low-risk, and medium-risk projects. There are pessimistic, most likely, and optimistic scenarios for the NPVs of these projects, which are realized with the probabilities shown in the table.**

Risk	Projects	Probability		
		Pessimistic	Most likely	Optimistic
No risk	2, 4, 5, 7, 9, 10, 12, 14	0	1	0
Low risk	3, 6, 15	1/6	2/3	1/6
Medium risk	1, 8, 11, 13, 16	1/3	1/2	1/6

A detailed solution to the 90-scenario model would be the analog of Table 19 with two differences: first there would be 90 columns, rather than 10 columns, detailing the scenarios of the budget and the pessimistic, most likely, and optimistic scenarios for the low-risk and medium-risk projects; and second instead of “1” and “0” we would have Plan “A”, “B”, “C” or “Do Nothing” to capture the timing options. For obvious reasons, we do not present this level of detail. However, Table 22 summarizes an optimal prioritization and shows the frequency—out of 90 scenarios—with which each project is implemented under each plan. While the 90 scenarios do not all have equal probability mass, the table suggests, in a simple and interpretable way, how higher priority projects are implemented more frequently under more preferred means. While there are exceptions (e.g., Project 13), medium risk projects tend to receive lower priority. Must-do projects are not necessarily top priority because their implementation can be delayed within the five-year horizon.

Table 22 shows a solution which is optimized with respect to the *expected* NPV, but the solution does not account for risk. What does risk mean in the context of capital budgeting under uncertainty? Figure 27 shows a histogram of the NPV values from the optimal prioritized solution, accounting for the probability mass associated with each scenario. We may be concerned about the low values of NPV in the left-hand tail, e.g., \$129M and \$144M. Would a different objective function that accounts for risk lead to a solution that, in some sense, better accounts for risk by decreasing the likelihood of low NPV realizations? In what follows we describe principled ways to optimize priorities in the context of capital budgeting while accounting for the risk of low-NPV outcomes.

**Table 22. Optimal prioritization when both budget, costs, and NPV values are uncertain, with a total of 90 scenarios. The “Priority” column indicates the priority level, ranging from 1-16. The columns for Plan A, B, C, and Do Nothing indicate the number of scenarios in which each option is selected. Note that while “must do” projects are selected under all 90 scenarios, their priority can decrease based on the timing of their implementation. The expected NPV associated with this prioritization is \$168.90M.**

ID	Project name	Category	Risk	Priority	Plan A	Plan B	Plan C	Do Nothing
12	Reactor vessel upgrade (head included)	Must do	No	1-5	90	0	0	0
14	Replace instrumentation and control cables	Must do	No	1-5	90	0	0	0
9	Batteries replacement	Optional	No	1-5	90	0	0	0
6	Seismic modification, requalification, reinforcement, improvement	Optional	Low	1-5	90	0	0	0
2	Pressurizer replacement	Must do	No	1-5	90	0	0	0
4	Secondary system PHM system	Optional	No	6	69	21	0	0
13	Replace LP turbine	Optional	Med	7-8	63	27	0	0
3	Improvement to emergency diesel generators	Optional	Low	7-8	63	27	0	0
1	HP feedwater heater upgrade	Optional	Med	9	27	63	0	0
5	Replacement of two reactor coolant pumps	Must do	No	10	12	78	0	0
7	Fire protection	Must do	No	11	6	84	0	0
10	Replace CCW piping, heat exchangers, valves	Must do	No	12	6	71	13	0
15	Condenser retubing	Optional	Low	13	0	77	0	13
16	Replace moisture separator reheater	Optional	Med	14	0	48	5	37
8	Service water system upgrade	Optional	Med	15	0	48	0	42
11	Reactor vessel internals	Optional	Med	16	0	18	0	72

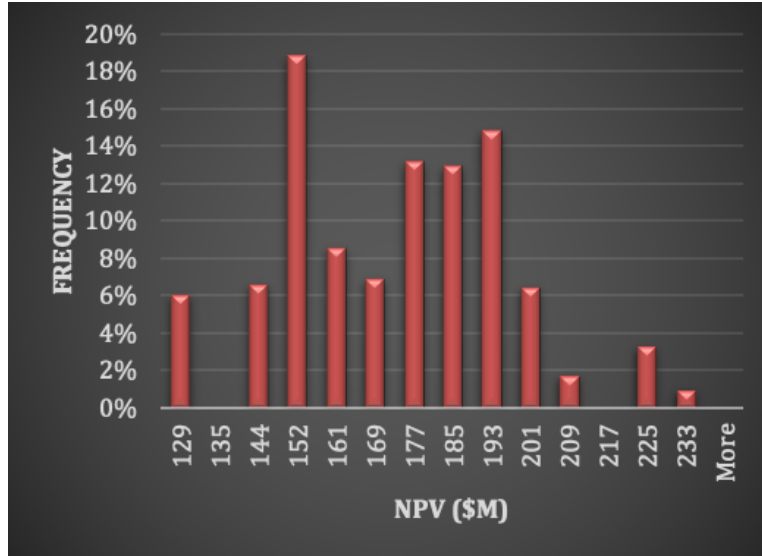


Figure 27. Histogram of NPV realizations across 90 scenarios accounting for the probability mass of each scenario. The optimized priority list maximizes the expected net present value but does not account for risk.

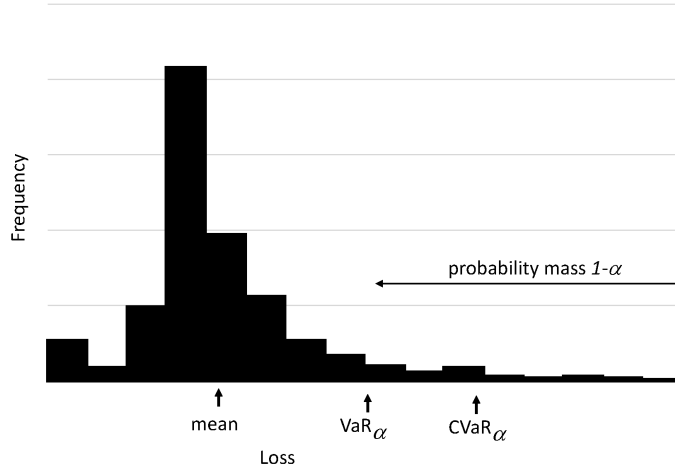
## 8.2 Risk-Based Stochastic Capital Budgeting using Conditional Value-at-Risk

### 8.2.1 Definitions of VaR and CVaR

Let  $X$  be a random variable with a cumulative distribution function  $F_X(z) = P\{X \leq z\}$ . It will be useful to think of  $X$  as a “loss,” or more generally be such that large values are bad. The VaR of  $X$  with confidence level  $\alpha$  (e.g.,  $\alpha = 0.90$ ) is:

$$VaR_\alpha(X) = \min\{z | F_X(z) \geq \alpha\}, \quad (13)$$

which is equivalent to  $VaR_\alpha(X) = F_X^{-1}(\alpha)$  if  $X$  is a continuous random variable. By this definition,  $VaR_\alpha(X)$  is a (lower)  $\alpha$ -percentile of the random variable  $X$ . An alternative measure of risk is conditional value-at-risk,  $CVaR_\alpha(X)$ , introduced [20]. Here,  $CVaR_\alpha(X)$  is the conditional expectation of  $X$  given that  $X \geq VaR_\alpha(X)$ . Figure 28 (see also [21]) shows the relationship between these two measures of risk,  $VaR_\alpha(X)$  and  $CVaR_\alpha(X)$ .



**Figure 28. Relationship between value-at-risk and conditional value-at-risk. A typical value of  $\alpha$  is  $\alpha=0.90$ .**

The typical definition of  $CVaR_\alpha(X)$  is  $CVaR_\alpha(X) = \mathbb{E}\{X|X > VaR_\alpha(X)\}$ . There are alternative ways to define this measure, which are mathematically equivalent. Rockafellar and Uryasev in [20] (see also [22]) defines CVaR as:

$$CVaR_\alpha[X] = \min_u \left\{ u + \frac{1}{1-\alpha} \mathbb{E}[X - u]^+ \right\}, \quad (14)$$

where  $[X - u]^+ = \max(X - u, 0)$ . Here, variable  $u$  is simply an auxiliary decision variable whose optimal value turns out to be  $VaR_\alpha(X)$ . The above definition is particularly useful for computation in the context of optimization.

Researchers have argued for using CVaR over VaR as a measure of risk. Theoretically, CVaR satisfies the assumptions of a so-called coherent risk measure (see [23]), and VaR does not. In simpler terms, minimizing VaR is concerned with the numerical value of the 95-th percentile (say) of the loss, but it does not care about the magnitude of larger losses. CVaR takes these magnitudes into account.

## 8.2.2 CVaR in Capital Budgeting

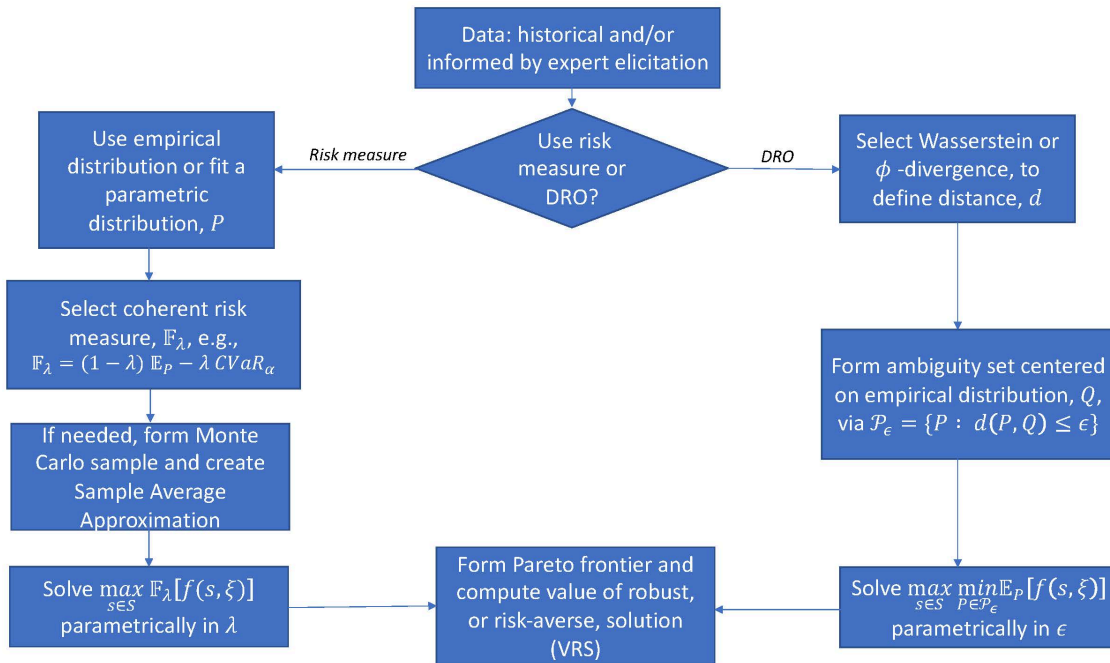
Figure 29 is a flow-chart giving a roadmap for the model development and analysis that follows. While our focus is on capital budgeting, the framework applies more broadly. We start with data, which may be historical and believed to be representative, or plausible scenarios derived from expert elicitation. In what follows we first consider the left-hand path in the flow-chart. That means that we construct an explicit risk measure, and in particular, we will use a weighted combination of expectation and CVaR, which we discuss in detail below.

This approach allows us to parametrically vary the weight on maximizing expected NPV versus penalizing solutions that yield low-NPV scenarios, and we denote the weight by  $\lambda$  with  $0 \leq \lambda \leq 1$ . Let  $NPV(s, \xi)$  denote the net present value under a prioritization decision specified by decision  $s$ , and under a realization of the budget and profit of each project, denoted by  $\xi$ . Then, we seek to solve the following optimization model:

$$\max_{s \in S} (1 - \lambda)\mathbb{E}[NPV(s, \xi)] - \lambda CVaR_\alpha[-NPV(s, \xi)]. \quad (15)$$

When  $\lambda = 0$  the model reduces to that discussed above; i.e., we seek a prioritization decision,  $s$ , to maximize expected NPV, where “ $s \in S$ ” simply indicates the constraints that a prioritized solution must satisfy.

As just discussed above,  $CVaR_\alpha[X]$  is typically applied to a random variable,  $X$ , which represents a loss; i.e., we seek to avoid large values of  $X$ . In this context, let  $VaR_\alpha[X]$  denote the  $\alpha$ -level quantile of  $X$ . Thus, if  $\alpha = 0.75$  then  $VaR_{0.75}[X]$  is the value such that 75% of the realizations of  $X$  have lower values of loss. Suppose for simplicity that  $NPV(s, \xi)$  values are positive. Large values of  $NPV(s, \xi)$  are good, and hence large values of  $-NPV(s, \xi)$  (i.e., those closer to zero) are bad. In Figure 27 large values of  $-NPV(s, \xi)$  are, within a sign, \$129M and \$144M. Using the definition of  $CVaR_\alpha[X] = \mathbb{E}[X | X > VaR_\alpha[X]]$  we thus have that the conditional value-at-risk is the expected value of the loss, given that the loss exceeds a certain percentile. So, when  $\lambda = 1$  we seek to minimize the expected value of NPV given that they fall below a threshold. More generally, values of  $\lambda$  between 0 and 1 seek a trade-off between reward and risk, captured by expected NPV and CVaR, respectively.



**Figure 29. Flow-chart illustrating two principled ways to approach risk-averse optimization. In both cases we start with available data. The left path then fits a probability distribution to that data, and then select a risk-averse objective function. One popular choice is a weighted sum of expectation and conditional value-at-risk. We then minimize risk, after possibly forming a Monte Carlo sample average approximation or employing some other discretization of the probability distribution. The right-hand path instead formulates and solves a distributionally robust optimization model. We defer detailed discussion of the right-hand path to Section 11.**

### 8.2.3 CVaR Mathematical Optimization Model

We briefly state the CVaR-based optimization model for completeness. The basic notation and formulation—albeit without the CVaR component—are presented in detail in [19]. The following model

combines three sub-models: (i) the basic knapsack-style optimization model for deterministic capital budgeting, (ii) the prioritization model, and (iii) optimization-based formulation of conditional value of risk with auxiliary variable  $u$ .

The objective function in (CVaR-1a) forms a weighted sum of NPV and CVaR, using respective weights  $1 - \lambda$  and  $\lambda$ . Constraint (CVaR-1b) linearizes the positive-part operator. Constraint (CVaR-1c) says that either project  $i$  is higher priority than  $i'$  or vice versa, allowing for ties. Constraint (CVaR-1d) requires that if project  $i$  is higher priority than  $i'$  ( $s_{ii'} = 1$ ) and we select the lower priority project ( $y_{i'}^\omega = 1$ ) then we must also select the higher priority project ( $y_i^\omega = 1$ ) all under scenario  $\omega$ . Constraint (CVaR-1e) requires that we respect annual capital budgets. Constraint (CVaR-1f) determines whether we select project  $i$  under some plan ( $y_i^\omega = 1$ ) or whether we choose the do-nothing option ( $y_i^\omega = 0$ ). Constraint (CVaR-1g) says that “do-nothing” is not an option for must-do projects. Constraint (CVaR-1h) is a piggy-backing constraint, which does not occur in our problem instance. Constraints (CVaR-1i) and (CVaR-1j) are logical constraints concerning a full prioritization to avoid illegal “cycles” in the logic, e.g., avoiding that Project 3 is higher priority than Project 2, which is higher priority than Project 7, which is higher priority than Project 3. Constraint (CVaR-1k) says that if project  $i$  is higher priority than  $i'$  ( $s_{ii'} = 1$ ) and the lower priority project is selected via Plan B (say) then the higher priority project  $i$  can only be selected via Plan A or B (but not C) under that scenario.

$$\max_{s,x,y,z,v,u} (1 - \lambda) \sum_{\omega \in \Omega} q^\omega \sum_{i \in I} \sum_{j \in J_i} a_{ij}^\omega x_{ij}^\omega - \lambda \left[ u + \frac{1}{1 - \alpha} \sum_{\omega \in \Omega} q^\omega v^\omega \right] \quad (\text{CVaR-1a})$$

$$\text{s.t.} \quad v^\omega \geq - \sum_{i \in I} \sum_{j \in J_i} a_{ij}^\omega x_{ij}^\omega - u, \omega \in \Omega \quad (\text{CVaR-1b})$$

$$s_{ii'} + s_{i'i} \geq 1, i < i', i, i' \in I \quad (\text{CVaR-1c})$$

$$y_i^\omega \geq y_{i'}^\omega + s_{ii'} - 1, i \neq i', i, i' \in I, \omega \in \Omega \quad (\text{CVaR-1d})$$

$$\sum_{i \in I} \sum_{j \in J_i} c_{ijkt}^\omega x_{ij}^\omega \leq b_{kt}^\omega, \quad k \in K, t \in T, \omega \in \Omega \quad (\text{CVaR-1e})$$

$$\sum_{j \in J_i} x_{ij}^\omega = y_i^\omega, \quad i \in I, \omega \in \Omega \quad (\text{CVaR-1f})$$

$$y_i^\omega = 1, \quad i \in M, \omega \in \Omega \quad (\text{CVaR-1g})$$

$$x_{i'j'}^\omega \leq x_{ij}^\omega, \quad (i', j') \in IJ_{ij}, j \in J_i, i \in I \quad (\text{CVaR-1h})$$

$$s_{ii'} + s_{i'i''} + s_{i''i} \leq 2, \quad i \neq i', i' \neq i'', i'' \neq i, i, i', i'' \in I \quad (\text{CVaR-1i})$$

$$s_{ii'} + s_{i'i} \leq 1, \quad i < i', i, i' \in I \quad (\text{CVaR-1j})$$

$$x_{i'j}^\omega + s_{ii'} - 1 \leq \sum_{\substack{j' \in J_i \\ j' \leq j}} x_{ij'}^\omega, \quad i \neq i', i, i' \in I, j \in J_{i'}, \omega \in \Omega \quad (\text{CVaR-1k})$$

$$s_{ii'}, x_{ij}^\omega, y_i^\omega \in \{0, 1\}, i \neq i', i, i' \in I, j \in J_i, \omega \in \Omega \quad (\text{CVaR-1l})$$

$$v^\omega \geq 0, \omega \in \Omega. \quad (\text{CVaR-1m})$$

## 8.2.4 Analysis of Risk Versus Return Optimization using CVaR and NPV

We now revisit the analysis presented above in the risk-neutral case (i.e.,  $\lambda = 0$ ) by showing an optimal prioritization using input data from Table 20. We use  $\alpha = 0.75$  so that we are concerned about the lowest

25% of NPV realizations in terms of CVaR. With this value of  $\alpha$ , Table 23 shows an optimal prioritization for  $\lambda = 0, 0.5, \text{ and } 1$ . As Table 19 and Table 22 suggest, there are often ties at certain prioritization levels, and here we break these ties arbitrarily, and this theme recurs in our analysis. The table shows that as the risk aversion parameter  $\lambda$  grows, most of the projects see little movement in the priority list. However, Project 2 drops in priority from being ranked 5<sup>th</sup> to 11<sup>th</sup>, Project 3 drops from being ranked 8<sup>th</sup> to 13<sup>th</sup>, and Project 15 climbs in the prioritization from 13<sup>th</sup> to 4<sup>th</sup>.

At  $\lambda = 0$  optional Project 3 – which has negative NPV under the pessimistic low-risk profit scenarios – is implemented under Plan A in 63 scenarios and under Plan B in 27 scenarios. At  $\lambda = 1$  its lower prioritization has it implemented under Plan B in 61 scenarios, and it is not selected in 29 scenarios, avoiding the negative NPV realizations, and reducing down-side risk. In concert with this change, we delay the must-do Project 2 from implementation under Plan A in all 90 scenarios when  $\lambda = 0$  to implementation under Plan B in 51 scenarios and Plan C in 39 scenarios. The altered timing of the associated costs for Projects 2 and 3 frees up first-year budget to select Project 15 under Plan A in all 90 scenarios under  $\lambda = 1$  rather than under Plan B in 77 scenarios and not selected under 13 scenarios when  $\lambda = 0$ .

**Table 23. Optimal project prioritization. Here we optimize a weighted sum of the expected value of NPV (weight  $1-\lambda$ ) and CVaR (weight  $\lambda$ ). The respective NPV and CVaR values are as follows:  $\lambda=0$ : (NPV=168.90, CVaR=140.54),  $\lambda=0.5$ : (NPV=167.68, CVaR=142.72), and  $\lambda=1$ : (NPV=166.51, CVaR=143.05). We note that we write  $\lambda=0^+$  and  $\lambda=1^-$  in the column headers because we prefer to place small positive weight on CVaR in the former case and NPV in the latter case. For example, solving with  $\lambda=0.0$  yields the same NPV but CVaR of 128.16, a dramatic increase in risk, while using  $\lambda=0.01$  or  $0.05$  results in the same NPV but eliminates low NPV scenarios.**

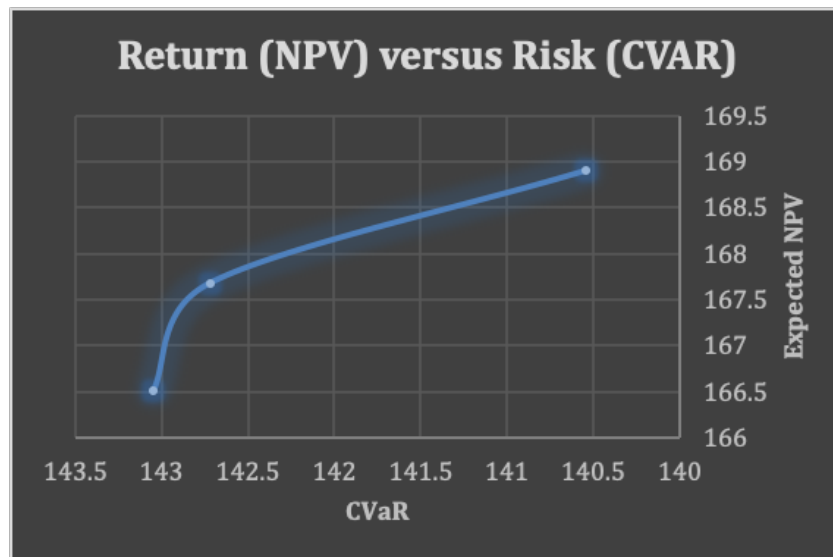
ID	Project name	Category	Risk	$\lambda = 0^+$	$\lambda = 0.5$	$\lambda = 1^-$
1	HP feedwater heater upgrade	Optional	Med	9	8	8
2	Pressurizer replacement	Must do	No	5	11	11
3	Improvement to emergency diesel generators	Optional	Low	8	15	13
4	Secondary system PHM system	Optional	No	6	6	5
5	Replacement of two reactor coolant pumps	Must do	No	10	9	10
6	Seismic modification, requalification, reinforcement, improvement	Optional	Low	4	7	6
7	Fire protection	Must do	No	11	10	9
8	Service water system upgrade	Optional	Med	15	14	15
9	Batteries replacement	Optional	No	3	3	3
10	Replace CCW piping, heat exchangers, valves	Must do	No	12	12	12
11	Reactor vessel internals	Optional	Med	16	16	16
12	Reactor vessel upgrade (head included)	Must do	No	1	1	1
13	Replace LP turbine	Optional	Med	7	5	7
14	Replace instrumentation and control cables	Must do	No	2	2	2
15	Condenser retubing	Optional	Low	13	4	4
16	Replace moisture separator reheater	Optional	Med	14	13	14



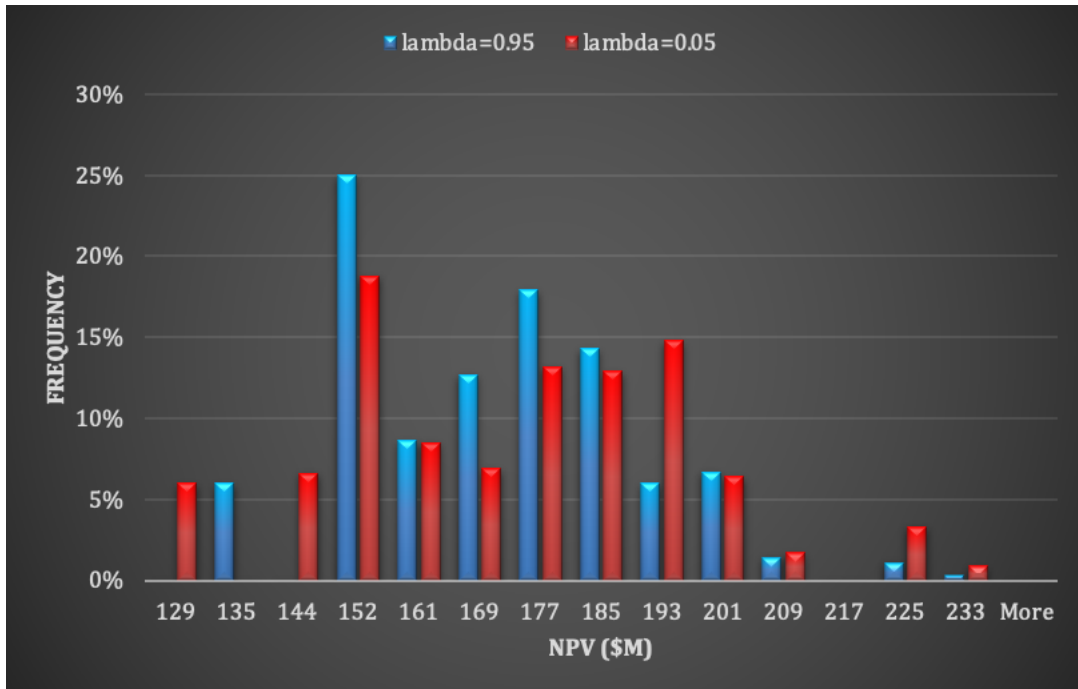
Figure 30 shows the Pareto efficient frontier representing the tradeoff between risk (CVaR on the horizontal axis, with risk growing left-to-right, as is standard in such plots) and return (expected NPV growing on the vertical axis). The upper-right point corresponds to  $\lambda = 0$  in which we maximize expected NPV. As we move from  $\lambda = 0$  to  $\lambda = 0.5$ , the expected NPV drops by \$1.22M in order to reduce risk as measured by CVaR by \$2.18M. An additional reduction in risk of \$0.33M is possible but requires reducing NPV by an additional \$1.17M.

Figure 31 allows us to further visualize risk reduction by plotting the probability distribution of NPVs across the 90 scenarios, weighted by their respective probabilities. The red bars correspond to  $\lambda = 0.05$  when most weight is on maximizing NPV and the blue histogram corresponds to  $\lambda = 0.95$  when most weight corresponds to minimizing CVaR to reduce low NPV realizations. As the weight on the risk term grows, note how the distribution of the NPV values change. The latter risk-averse solution increases the lowest NPV realization of \$128M (\$129M bin) by \$4M to \$132M (\$135 bin) with a probability mass of 6%.

Similarly, the risk-averse solution effectively increases \$144M (7%) by \$8M to \$152M. All subsequent realizations reading left-to-right come in pairs. For example, the 25% blue bar and 19% red bar both correspond to the \$152M bin, and the 9% bars both correspond to the \$161M bin. The value of  $\alpha$  is 0.75, so can observe that the 25<sup>th</sup> percentile (value of \$152M) is the concentration point. Overall, the risk-averse histogram reduces the left-hand tail and the expense of reducing the overall expected NPV.



**Figure 30. Subset of the Pareto frontier using just  $\lambda=0,0.5$ , and 1, where the risk is measured by CVaR and return is measured by Expected NPV. Optimal solutions are obtained by solving model (1) with input from Table 17. The plot is oriented so that risk grows moving left-to-right on the x-axis and return grows moving bottom to top along the y-axis. Starting from the upper-right point, we can reduce risk by \$2.18M by reducing expected NPV by \$1.22M. Additional reduction is risk of \$0.33M requires reducing NPV by an additional \$1.17M.**



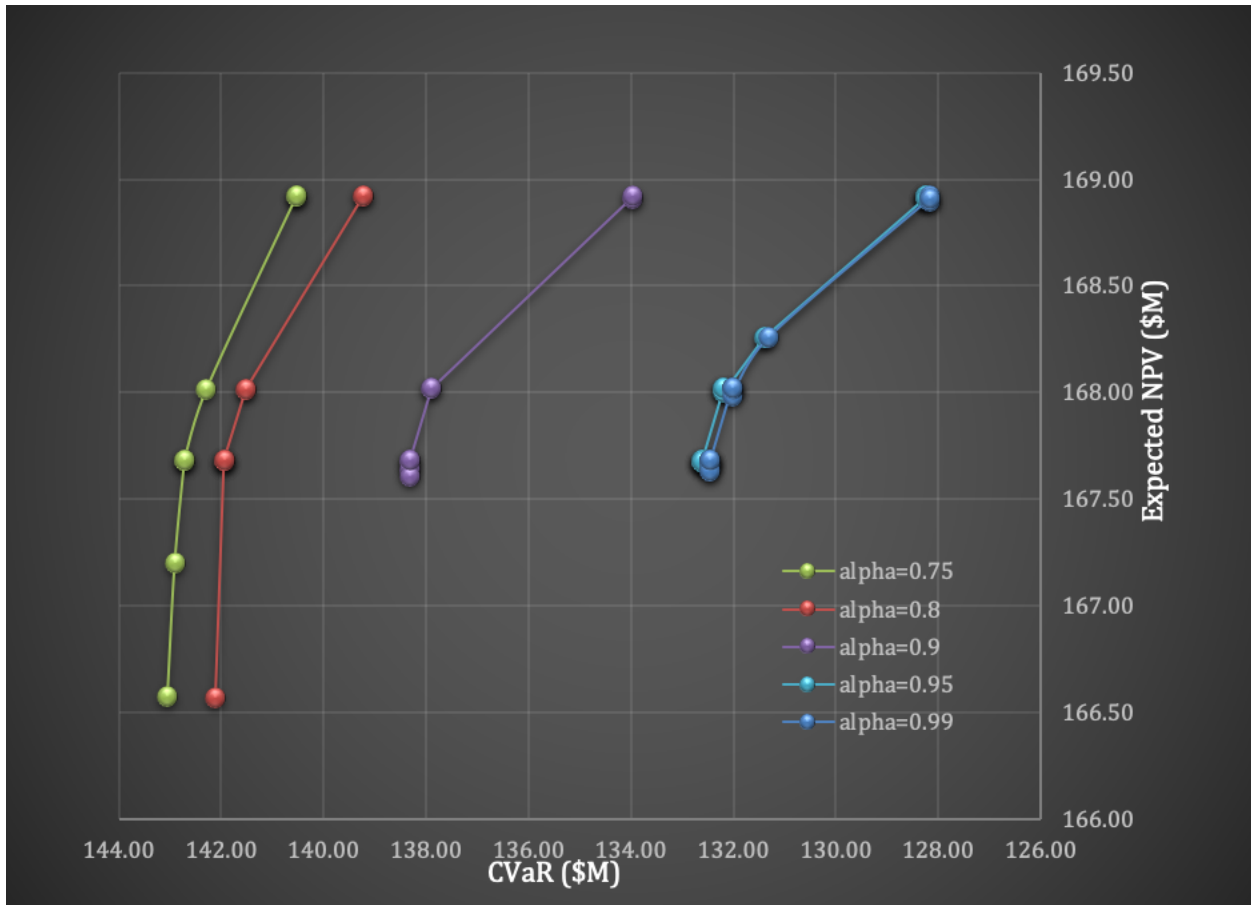
**Figure 31. Histogram for scenario NPVs accounting for probability mass associated with each scenario. The red histogram corresponds to  $\lambda=0.05$  when most weight is on maximizing NPV and the blue histogram corresponds to  $\lambda=0.95$  when most weight corresponds to minimizing CVaR for low NPV realizations.**

In the discussion above, we restricted attention to analysis with  $\alpha = 0.75$ , so that we were concerned about the lowest 25% of NPV realizations in computing risk via CVaR, and we further restricted attention to  $\lambda \approx 0$ ,  $\lambda = 0.5$  and  $\lambda \approx 1$  for the relative weight on risk (CVaR) and return (expected NPV). Here, we expand on that analysis. Figure 32 summarizes the risk-return tradeoff for  $\alpha = 0.75, 0.80, 0.90, 0.95$ , and  $0.99$ . The figure repeats the Pareto frontier for  $\alpha = 0.75$  (left-most plot) from Figure 30 except that now we include the full range of values of  $\lambda$  and achieve five rather than three points on the efficient frontier. The additional four plots repeat similar frontiers but for  $\alpha = 0.80, 0.90, 0.95, 0.99$ , where we focus on values that are further in the poor-outcome tail of NPV as  $\alpha$  grows.

For each plot, the top-right point is for low values of  $\lambda \approx 0$  (maximizing NPV) and the lower-left point on the frontier is for  $\lambda \approx 1$  (minimizing CVaR). Of course, the top-right points are identical in terms of NPV, regardless of the value of  $\alpha$ . As discussed above, for  $\alpha = 0.75$  moving from  $\lambda \approx 0$  to  $\lambda \approx 1$  results in the loss in the expected NPV of \$2.4M but a gain in CVaR of \$2.5M. As  $\lambda$  grows, the corresponding standard deviation of the NPV realizations, which is another measure of risk, decreases from \$22.9M to \$19.5M to \$18.9M as we move from  $\lambda \approx 0$  to  $\lambda \approx 0.5$  to  $\lambda \approx 1$ .

In other words, for  $\alpha = 0.75$  at the price of reducing expected NPV by \$2.4M we reduce risk, as measured by CVaR by \$2.5M, and as measured by standard deviation by \$4.0M. As another example, consider  $\alpha = 0.90$ . Here, we reduce risk as measured by CVaR by \$4.4M as we move from  $\lambda \approx 0$  to  $\lambda \approx 1$  corresponding to a drop in NPV of \$1.3M. The standard deviation in this case drops by \$3.5M from \$22.9M to \$19.4. It is not surprising that under  $\alpha = 0.90$  the standard deviation drops less than under  $\alpha = 0.75$  because CVaR only attempts to “shape” the 10% lower tail rather than the 25% lower tail of NPVs under  $\alpha = 0.75$ . In our view, these are risk-return tradeoffs worth considering.

More generally, Figure 32 shows diminishing returns; i.e., we obtain sharper drops in risk as we initially increase the weight on  $\lambda \approx 0$  to a small positive value. More modest reductions in risk follow, requiring greater loss of expected NPV. As the figure shows, there are specific NPV values that repeat across different values of  $\alpha$ , but as  $\alpha$  grows the conditional expectation increasingly focuses on conditional means of corresponding to lower realizations of NPV. The figure shows this via the Pareto frontiers moving to the right as the value of  $\alpha$  increases from 0.75 to 0.99; i.e., as we become more and more concerned about the tail.



**Figure 32. Pareto frontiers for different values of  $\alpha$ . The figure repeats the Pareto frontier for  $\alpha=0.75$  (left-most plot) from Figure 30 except that now we include the full range of values of  $\lambda$  and achieve five rather than three points on the efficient frontier. The additional four plots repeat similar frontiers but for  $\alpha=0.8, 0.9, 0.95, 0.99$ , where we focus on values that are further in the poor-outcome tail of NPV. Note that as the figure shows, there are specific NPV values that repeat across different values of  $\alpha$ , but as  $\alpha$  grows the conditional expectation increasingly focuses on low realizations.**

Table 24 shows optimal project ranking for  $\alpha = 0.75$  when solving model (1) for a range of values of  $\lambda$ . While there are other small changes, Table 24 a higher fidelity version of Table 23 and suggests that the trends discussed in the context of Table 23 capture the important shifts in prioritization as we become increasingly risk averse.

**Table 24. Optimal project prioritization from solving model (1) when  $\alpha=0.75$  and  $\lambda$  ranges from 0.1 to 0.9.**

ID	Project name	Category	Risk	$\lambda=0.1$	$\lambda=0.2$	$\lambda=0.3$	$\lambda=0.4$	$\lambda=0.5$	$\lambda=0.6$	$\lambda=0.7$	$\lambda=0.8$	$\lambda=0.9$
1	HP feedwater heater upgrade	Optional	Med	9	9	9	8	8	8	8	8	8
2	Pressurizer replacement	Must do	No	5	5	5	13	11	12	12	11	11
3	Improvement to emergency diesel generators	Optional	Low	8	8	8	15	15	15	15	14	13
4	Secondary system PHM system	Optional	No	6	6	6	6	6	6	7	5	5
5	Replacement of two reactor coolant pumps	Must do	No	10	10	10	11	9	10	10	10	10
6	Seismic modification, requalification, reinforcement, improvement	Optional	Low	4	4	4	7	7	7	6	6	6
7	Fire protection	Must do	No	11	11	11	10	10	9	9	9	9
8	Service water system upgrade	Optional	Med	15	15	15	14	14	14	14	15	15
9	Batteries replacement	Optional	No	3	3	3	3	3	3	3	3	3
10	Replace CCW piping, heat exchangers, valves	Must do	No	12	12	12	12	12	11	11	12	12
11	Reactor vessel internals	Optional	Med	16	16	16	16	16	16	16	16	16
12	Reactor vessel upgrade (head included)	Must do	No	1	1	1	1	1	1	1	1	1
13	Replace LP turbine	Optional	Med	7	7	7	5	5	5	5	7	7
14	Replace instrumentation and control cables	Must do	No	2	2	2	2	2	2	2	2	2
15	Condenser retubing	Optional	Low	13	13	13	4	4	4	4	4	4
16	Replace moisture separator reheater	Optional	Med	14	14	14	9	13	13	13	13	14
<b>NPV (\$M)</b>				168.92	168.92	168.92	168.02	167.68	167.68	167.68	167.20	166.57
<b>CVaR<sub>0.75</sub> (\$M)</b>				140.54	140.54	140.54	142.29	142.72	142.72	142.72	142.91	143.05
<b>Standard Deviation (\$M)</b>				22.90	22.90	22.90	19.89	19.52	19.52	19.49	19.11	18.82

In the remainder of Section 10, we repeat the weighted sum formulation shown in model (CVAR-1) for the optimal project prioritization problem to a range of related models, starting with the single knapsack model.

### 8.2.5 CVaR for Single Knapsack Problem

$$\max(1 - \lambda) \sum_{\omega \in \Omega} q^\omega \sum_{i \in N} p_i^\omega x_i^\omega - \lambda \left[ u + \frac{1}{1 - \alpha} \sum_{\omega \in \Omega} q^\omega v^\omega \right] \quad (\text{CVAR-2a})$$

$$v^\omega \geq - \sum_{i \in N} p_i^\omega x_i^\omega - u, \omega \in \Omega \quad (\text{CVAR-2b})$$

$$\sum_{i \in N} w_i^\omega x_i^\omega \leq c^\omega \quad (\text{CVAR-2c})$$

$$y_{i,i'} + y_{i',i} \geq 1 \text{ and } i < i' \quad (\text{CVAR-2d})$$

$$x_i^\omega \geq x_{i'}^\omega + y_{i,i'} - 1 \text{ and } i \neq i' \quad (\text{CVAR-2e})$$

$$x_i^\omega, y_{ii'}^\omega \in \{0,1\}, i \neq i', i, i' \in I, \omega \in \Omega \quad (\text{CVAR-2f})$$

$$v^\omega \geq 0, \omega \in \Omega, \lambda \in [0, 1] \quad (\text{CVAR-2g})$$

### 8.2.6 CVaR for Multi-Dimensional Knapsack Problem

$$\max(1 - \lambda) \sum_{\omega \in \Omega} q^\omega \sum_{i \in N} p_i^\omega x_i^\omega - \lambda \left[ u + \frac{1}{1 - \alpha} \sum_{\omega \in \Omega} q^\omega v^\omega \right] \quad (\text{CVAR-3a})$$

$$v^\omega \geq - \sum_{i \in N} p_i^\omega x_i^\omega - u, \omega \in \Omega \quad (\text{CVAR-3b})$$

$$\sum_{d \in D} w_{i,d}^\omega x_i^\omega \leq c_d^\omega \quad (\text{CVAR-3c})$$

$$y_{i,i'} + y_{i',i} \geq 1 \text{ and } i < i' \quad (\text{CVAR-3d})$$

$$x_i^\omega \geq x_{i'}^\omega + y_{i,i'} - 1 \text{ and } i \neq i' \quad (\text{CVAR-3e})$$

$$x_i^\omega, y_{ii'}^\omega \in \{0,1\}, i \neq i', i, i' \in I, \omega \in \Omega \quad (\text{CVAR-3f})$$

$$v^\omega \geq 0, \omega \in \Omega, \lambda \in [0, 1] \quad (\text{CVAR-3g})$$

### 8.2.7 CVaR for Multiple Knapsack Problem

$$\max(1 - \lambda) \sum_{\omega \in \Omega} q^\omega \sum_{m \in M} \sum_{i \in N} p_i^\omega x_{i,m}^\omega - \lambda \left[ u + \frac{1}{1 - \alpha} \sum_{\omega \in \Omega} q^\omega v^\omega \right] \quad (\text{CVAR-4a})$$

$$v^\omega \geq - \sum_{m \in M} \sum_{i \in N} p_i^\omega x_{i,m}^\omega - u, \omega \in \Omega \quad (\text{CVAR-4b})$$

$$\sum_{i \in N} w_i^\omega x_{i,m}^\omega \leq c_m^\omega \quad (\text{CVAR-4c})$$

$$y_{i,i'} + y_{i',i} \geq 1 \text{ and } i < i' \quad (\text{CVAR-4d})$$

$$\sum_{m \in M} x_{i,m}^\omega \geq \sum_{m \in M} x_{i',m}^\omega + y_{i,i'} - 1 \text{ and } i \neq i' \quad (\text{CVAR-4e})$$

$$\sum_{m \in M} x_{i,m}^\omega \leq 1 \quad (\text{CVAR-4f})$$

$$x_{i,m}^\omega, y_{ii'}^\omega \in \{0,1\}, i \neq i', i, i' \in I, \omega \in \Omega \quad (\text{CVAR-4g})$$

$$v^\omega \geq 0, \omega \in \Omega, \lambda \in [0, 1] \quad (\text{CVAR-4e})$$

### 8.2.8 CVaR for Multiple-Choice Knapsack Problem

$$\max(1 - \lambda) \sum_{\omega \in \Omega} q^\omega \sum_{j \in J_i} \sum_{i \in N} p_{i,j}^\omega x_{i,j}^\omega - \lambda \left[ u + \frac{1}{1 - \alpha} \sum_{\omega \in \Omega} q^\omega v^\omega \right] \quad (\text{CVAR-4a})$$

$$v^\omega \geq - \sum_{j \in J_i} \sum_{i \in N} p_{i,j}^\omega x_{i,j}^\omega - u, \omega \in \Omega \quad (\text{CVAR-4b})$$

$$\sum_{i \in N} w_{i,j}^\omega x_{i,j}^\omega \leq c^\omega \quad (\text{CVAR-4c})$$

$$y_{i,i'} + y_{i',i} \geq 1 \text{ and } i < i' \quad (\text{CVAR-4d})$$

$$\sum_{j=1}^{J_i-1} x_{i,j}^\omega \geq \sum_{j=1}^{J_{i'}-1} x_{i',j}^\omega + y_{i,i'} - 1 \text{ and } i \neq i' \quad (\text{CVAR-4e})$$

$$\sum_{j \in J_i} x_{i,j}^\omega = 1 \quad (\text{CVAR-4f})$$

$$x_{i,m}^\omega, y_{ii'}^\omega \in \{0,1\}, i \neq i', i, i' \in I, \omega \in \Omega \quad (\text{CVAR-4g})$$

$$v^\omega \geq 0, \omega \in \Omega, \lambda \in [0, 1] \quad (\text{CVAR-4h})$$

## 9. DISTRIBUTIONALLY ROBUST OPTIMIZATION

### 9.1 Overview

In Section 8 we considered a risk-averse approach to capital budgeting by optimizing a weighted sum of expected NPV and CVaR, following the left-hand path in Figure 29. We now consider the figure's right-hand path, i.e., we consider an approach to risk-averse decision making using DRO [24]. To this end, we begin with a *nominal* stochastic optimization problem:

$$\max_{s \in S} \sum_{\sigma \in \Sigma} q^\sigma f(s, \xi^\sigma) \quad (16)$$

In our context, the nominal model is a stochastic capital budgeting problem in which we maximize expected NPV and, following the notation in Section 10, we could have  $f(s, \xi^\sigma) = NPV(s, \xi^\sigma)$ . In this context “ $s \in S$ ” simply indicates the constraints that a prioritized solution must satisfy, wherein we prioritize project selection subject to uncertainty in costs and the NPV of each project as well as uncertainty in resource availability. The goal is to prioritize so as to maximize expected NPV, assuming that the nominal distribution, specified by the probability mass function  $q^\sigma$ ,  $\sigma \in \Sigma$ , is correct.

We suppose that  $\xi$  is a discrete random variable with finite sample space  $\Omega$ , so that  $\xi^\omega$ ,  $\omega \in \Omega$ , enumerate all possible realizations. We further suppose that we only have observations of  $\xi^\sigma$ , for  $\sigma \in \Sigma \subset \Omega$ , i.e., possibly a strict subset, which may arise in a data-driven setting. In such a data-driven setting we could have, for example, probability mass  $q^\sigma = 1/|\Sigma|$  for all  $\sigma \in \Sigma$ .

A DRO variant of this nominal stochastic optimization model is then given by:

$$\max_{s \in \mathcal{S}} \min_{p \in P} \sum_{\omega \in \Omega} p^\omega f(s, \xi^\omega). \quad (17)$$

Here, we may view this DRO model as playing a “game” against nature. First, we select  $s \in \mathcal{S}$ , and then knowing  $s$ , nature selects a worst-case probability distribution,  $p \in P$ , to minimize the expected NPV, which we seek to maximize. We will make precise what we mean by the Distributional Uncertainty Set (DUS), denoted by  $P$ , below, but for the moment it is enough to think of the set as representing a neighborhood of probability distributions centered on the given probability mass function,  $q$ , with the radius of the neighborhood specified by parameter  $\varepsilon$ . If  $\varepsilon = 0$  then the DRO model reduces to the nominal model. If  $\varepsilon$  is very large then nature will select the single worst-case scenario, e.g., the scenario with lowest budgets, highest costs, and lowest NPVs. This is too conservative to be useful (i.e. if we are living in this world, it is very likely that the plant would be uneconomical no matter what decisions are made). However, with moderate values of  $\varepsilon$  we obtain solutions that hedge against deviations from  $q$  without being excessively conservative.

Importantly, we do not view nature as malevolent, despite occasional evidence to the contrary. Rather, we use “min” to combat over-adapting our solution to a specific assumption about the probability distribution. In this sense, DRO plays the role of a “regularizer” to combat over-fitting that is analogous to regularizers used in high-dimensional statistics and statistical machine learning. Our approach to DRO requires further mathematical and intuitive development before we can analyze solutions to the DRO problem and compare them with the CVaR approach.

## 9.2 Defining a Distributional Uncertainty Set via the Wasserstein Distance

By the constraints denoted by  $p \in P$  we require that nature select a probability mass function,  $p$ , that is “close” to the nominal or empirical data-driven distribution  $q$ . We effect this by defining the DUS:

$$P = \left\{ p : D(p, q) \leq \varepsilon, \sum_{\omega \in \Omega} p^\omega = 1, p^\omega \geq 0, \omega \in \Omega \right\}, \quad (18)$$

where  $D(p, q)$  is the distance between nature’s choice,  $p$ , and the nominal data-driven distribution,  $q$ . As indicated above, the radius parameter  $\varepsilon$  governs the latitude we give nature, which in turn governs the degree of conservatism that we face when selecting decision  $s \in \mathcal{S}$ .

There are multiple ways to measure the “distance”,  $D(p, q)$ , between two probability distributions, which include the Kolmogorov-Smirnov distance, Kullback-Leibler divergence, chi-squared distances, total variation, and more general  $\phi$ -divergences. The Wasserstein distance, which is based on the idea of optimal transport, is a particularly useful way to measure such a distance in the context of distributionally robust optimization. For a distribution with known finite support, the Wasserstein distance,  $D(q, p)$ , between a given distribution,  $q^\sigma, \sigma \in \Sigma$ , and a given candidate robust distribution,  $p^\omega, \omega \in \Omega$ , is provided by the optimal value of the transportation problem:

$$\begin{aligned} D(q, p) = \min_z \quad & \sum_{\sigma \in \Sigma, \omega \in \Omega} d_{\sigma, \omega} z_{\sigma, \omega} \\ \text{s.t.} \quad & \sum_{\omega \in \Omega} z_{\sigma, \omega} = q^\sigma, \sigma \in \Sigma \end{aligned} \quad (19)$$

$$\sum_{\sigma \in \Sigma} z_{\sigma, \omega} = p^{\omega}, \omega \in \Omega$$

$$z_{\sigma, \omega} \geq 0, \sigma \in \Sigma, \omega \in \Omega.$$

The intuition behind this measure concerns the magnitude of probability mass,  $q^{\sigma}$ , that must be transported distance  $d_{\sigma, \omega}$  from vector  $\xi^{\sigma}$  to vector  $\xi^{\omega}$  via variable  $z_{\sigma, \omega}$ . At one extreme case, if the two sample spaces and probability mass functions coincide, i.e.,  $\Omega = \Sigma$  and  $p^{\omega} = q^{\omega}$ , and  $d_{\omega, \omega} = 0$  for all  $\omega \in \Omega$  then  $D(q, p) = 0$ .

To fully specify  $D(q, p)$  we must define  $d_{\sigma, \omega} = \text{dist}(\xi^{\sigma}, \xi^{\omega}), \sigma \in \Sigma, \omega \in \Omega$ . To do so we can select  $\text{dist}(\cdot, \cdot)$ , for example, to be the two-norm distance, or a more general  $\eta$ -norm distance, between the vectors,  $\xi^{\sigma}$  and  $\xi^{\omega}$ , i.e.,  $\text{dist}(\cdot, \cdot) = \|\xi^{\sigma} - \xi^{\omega}\|_{\eta}$ .

With the Wasserstein distance, if we are given distribution,  $q$ , we can then define:

$$P = \left\{ p : D(p, q) \leq \varepsilon, \sum_{\omega \in \Omega} p^{\omega} = 1, p^{\omega} \geq 0, \omega \in \Omega \right\} \quad (20)$$

for a given radius  $\varepsilon$ . Here, we think of  $P$  as a ball, or neighborhood, of probability distributions centered on  $q$ , where the neighborhood has radius  $\varepsilon$ . With  $\Sigma \subset \Omega$ , if  $\varepsilon = 0$  then  $P$  is the singleton  $\{q\}$ , and larger values of  $\varepsilon$  lead to increasingly large neighborhoods. In the context of robust optimization, if  $\varepsilon = 0$  then we will simply be solving the nominal stochastic optimization model, and as  $\varepsilon$  grows large we will consider increasingly conservative models.

We can now represent the set  $P$  via the following so-called extended-variable set of constraints:

$$\sum_{\sigma \in \Sigma, \omega \in \Omega} d_{\sigma, \omega} z_{\sigma, \omega} \leq \varepsilon \quad (21\text{-a})$$

$$\sum_{\omega \in \Omega} z_{\sigma, \omega} = q^{\sigma}, \sigma \in \Sigma \quad (21\text{-b})$$

$$\sum_{\sigma \in \Sigma} z_{\sigma, \omega} - p^{\omega} = 0, \omega \in \Omega \quad (21\text{-c})$$

$$z_{\sigma, \omega} \geq 0, \sigma \in \Sigma, \omega \in \Omega \quad (21\text{-d})$$

In other words, we can formalize the DUS via  $P = \{ p : \exists z \text{ satisfying } (21\text{-a}) - (21\text{-d}) \}$ .

### 9.3 Towards a Computationally Tractable Reformulation

Due to the  $\max_{s \in S} \min_{p \in P}$  construct in the DRO model, the model is not amenable to direct solution via optimization software. So, we reformulate the model to facilitate computation. For the moment let  $s \in S$  be fixed so that  $f(s, \xi^{\omega})$  is just a known numerical value for each  $\omega \in \Omega$ . Then, nature's problem may be written:



$$\min_{p,z} \sum_{\omega \in \Omega} p^\omega f(s, \xi^\omega) \quad (\text{DRO1-a})$$

$$\text{s.t.} \quad \sum_{\sigma \in \Sigma, \omega \in \Omega} d_{\sigma,\omega} z_{\sigma,\omega} \leq \varepsilon : [-\gamma] \quad (\text{DRO1-b})$$

$$\sum_{\omega \in \Omega} z_{\sigma,\omega} = q^\sigma, \sigma \in \Sigma : [v^\sigma] \quad (\text{DRO1-c})$$

$$\sum_{\sigma \in \Sigma} z_{\sigma,\omega} - p^\omega = 0, \omega \in \Omega : [\beta^\omega] \quad (\text{DRO1-d})$$

$$z_{\sigma,\omega} \geq 0, \sigma \in \Sigma, \omega \in \Omega. \quad (\text{DRO1-e})$$

Here,  $\gamma, v^\sigma, \beta^\omega$  denote dual variables. Note that in formulation (19),  $q$  and  $p$  are given, and  $z$  is the only free variable. In the DRO1 model (DRO1-a through DRO1-e),  $q^\sigma$  and  $\xi^\omega \forall \omega \in \Omega, \sigma \in \Sigma$  are given, and  $s$  is (temporarily) given, too, and in the model, nature optimizes over  $z$  and over  $p$  to select a worst-case distribution within radius  $\varepsilon$  of  $q$ .

Taking the dual of the linear program DRO1, and substituting out the dual variable  $\beta^\omega = -f(s, \xi^\omega)$ , we obtain the following:

$$\max_{\gamma, v} -\gamma\varepsilon + \sum_{\sigma \in \Sigma} v^\sigma q^\sigma \quad (\text{DRO2-a})$$

$$\text{s.t.} \quad -\gamma d_{\sigma,\omega} + v^\sigma \leq f(s, \xi^\omega), \sigma \in \Sigma, \omega \in \Omega \quad (\text{DRO2-b})$$

$$\gamma \geq 0 \quad (\text{DRO2-c})$$

To gain intuition regarding model (3), consider two extreme cases,  $\varepsilon = 0$  and  $\varepsilon = \infty$ . If  $\varepsilon = 0$  then there is no penalty in the objective function for allowing  $\gamma$  to grow large. As  $\gamma$  grows large, constraint (DRO2-b) becomes vacuous for all  $\sigma \neq \omega$ ; however, for  $\sigma = \omega$  we have  $d_{\sigma,\sigma} = \|\xi^\sigma - \xi^\omega\| = 0$ , and hence the constraint reduces to  $v^\sigma \leq f(s, \xi^\sigma)$ , and coupled with the objective function the optimal value reduces to  $\sum_{\sigma \in \Sigma} q^\sigma f(s, \xi^\sigma) = \sum_{\omega \in \Omega} p^\omega f(s, \xi^\omega)$ , i.e., it reduces to the objective function value of the nominal stochastic optimization model, as it must with  $\varepsilon = 0$ .

In the other extreme, as  $\varepsilon$  grows sufficiently large we must have  $\gamma = 0$  to avoid a huge penalty in the objective function. Thus, for each  $\sigma \in \Sigma$ , constraint (DRO2-b) reduces to  $v^\sigma \leq f(s, \xi^\omega), \forall \omega \in \Omega$ , i.e.,  $v^\sigma = \min_{\omega \in \Omega} f(s, \xi^\omega)$  so that  $v^\sigma$  takes on the lowest NPV for all  $\sigma$ , and the objective function reduces to:

$$\sum_{\sigma \in \Sigma} \min_{\omega \in \Omega} f(s, \xi^\omega) q^\sigma = \min_{\omega \in \Omega} f(s, \xi^\omega) \sum_{\sigma \in \Sigma} q^\sigma = \min_{\omega \in \Omega} f(s, \xi^\omega) \quad (\text{DRO3-a})$$

Again, this matches what it must: if  $\varepsilon = \infty$  then nature has enough latitude to place probability one on the single worst-case scenario.

## 9.4 A Computationally Tractable Reformulation

In Section 9.3 we fixed the primary decision variables,  $s$ , and took the dual of problem (17). The reason for doing so was to overcome the max-min structure in the DRO model. With the inner min reformulated as a max we can now formulate a single large maximization problem as follows:

$$\max_{\gamma, v} -\gamma\varepsilon + \sum_{\sigma \in \Sigma} v^\sigma q^\sigma \quad (\text{DRO4-a})$$

$$\text{s.t. } s \in S \quad (\text{DRO4-a})$$

$$-\gamma d_{\sigma, \omega} + v^\sigma \leq f(s, \xi^\omega), \sigma \in \Sigma, \omega \in \Omega \quad (\text{DRO4-a})$$

$$\gamma \geq 0 \quad (\text{DRO4-a})$$

Model (DRO4) provides the general formulation of a distributionally robust optimization model, which applies to any stochastic optimization model of our original nominal form. However, often  $f(s, \xi^\omega)$  and  $s \in S$  are shorthand for constructs in another model, and so in the next section we specify this for the stochastic capital budgeting model.

## 9.5 Tractable Reformulation Specialized to Stochastic Capital Budgeting

Specializing  $s \in S$  to be the constraints for prioritization, and specializing  $f(s, \xi^\omega)$  to define the NPV under scenario  $\omega \in \Omega$ , the DRO variant of the stochastic capital budgeting problem is as follows:

$$\max_{s, x, y, z, \gamma, v} -\gamma\varepsilon + \sum_{\sigma \in \Sigma} v^\sigma q^\sigma \quad (\text{DRO5-a})$$

$$\text{s.t. } -\gamma d_{\sigma, \omega} + v^\sigma \leq \sum_{i \in I} \sum_{j \in J_i} a_{ij}^\omega x_{ij}^\omega, \sigma \in \Sigma, \omega \in \Omega \quad (\text{DRO5-b})$$

$$s_{ii'} + s_{i'i} \geq 1, i < i', i, i' \in I \quad (\text{DRO5-c})$$

$$y_i^\omega \geq y_{i'}^\omega + s_{ii'} - 1, \quad i \neq i', i, i' \in I, \omega \in \Omega \quad (\text{DRO5-d})$$

$$\sum_{i \in I} \sum_{j \in J_i} c_{ijkt}^\omega x_{ij}^\omega \leq b_{kt}^\omega, \quad k \in K, t \in T, \omega \in \Omega \quad (\text{DRO5-e})$$

$$\sum_{j \in J_i} x_{ij}^\omega = y_i^\omega, \quad i \in I, \omega \in \Omega \quad (\text{DRO5-f})$$

$$y_i^\omega = 1, \quad i \in M, \omega \in \Omega \quad (\text{DRO5-g})$$

$$x_{i'j'}^\omega \leq x_{ij}^\omega, \quad (i', j') \in IJ_{ij}, j \in J_i, i \in I \quad (\text{DRO5-h})$$

$$s_{ii'} + s_{i'i''} + s_{i''i} \leq 2, \quad i \neq i', i' \neq i'', i'' \neq i, i, i', i'' \in I \quad (\text{DRO5-i})$$

$$s_{ii'} + s_{i'i} \leq 1, \quad i < i', i, i' \in I \quad (\text{DRO5-j})$$

$$x_{i'j}^\omega + s_{ii'} - 1 \leq \sum_{\substack{j' \in J_i \\ j' \leq j}} x_{ij}^\omega, \quad i \neq i', i, i' \in I, j \in J_{i'}, \omega \in \Omega \quad (\text{DRO5-k})$$

$$s_{ii'}, x_{ij}^\omega, y_i^\omega \in \{0,1\}, i \neq i', i, i' \in I, j \in J_i, \omega \in \Omega \quad (\text{DRO5-l})$$

$$\gamma \geq 0 \quad (\text{DRO5-m})$$

The decision variables and constraints of this model mimic those of the risk-averse NPV-CVaR formulation from Section 8.2, and so we do not repeat that discussion here. We note only that the objective function and the first constraint replicate their counterparts, (CVaR-a) and (CVaR-c), from the generic formulation of the DRO model.

## 9.6 Analysis of Risk versus Return Optimization using Distributionally Robust Optimization

Table 25 summarizes our results for DRO of the stochastic capital budgeting problem. As the value of  $\varepsilon$  grows from 0 to 1000 we become increasingly risk averse. The results are to be compared with Table 24 for our analysis using a weighted sum of NPV and CVaR.

Comparing the optimized prioritized rankings under DRO versus a weighted sum of NPV and CVaR, we see remarkable similarities in the results. As was the case for NPV-CVaR, Projects 2 and 3 fall in the rankings as risk aversion grows, and Project 15 climbs in prioritization. There are other shifts in the priority of individual projects as risk aversion grows, but they are more modest. Importantly we note that there can be multiple optimal solutions and thus sometimes small spurious differences arise, both between results reported here versus in Section 8 and within Table 25 as  $\varepsilon$  is varied parametrically.

The last eight rows of Table 25 display numerical performance measures. We emphasize that even though nature alters the probability distribution in the game-theoretic model, we report all results (expectation, standard deviation, and CVaR) with respect to the nominal probability distribution. We do this for two reasons. Foremost, as indicated above, we do not view nature as malevolent, and we are employing DRO to hedge against uncertainty. Second, reporting with respect to the nominal distribution allows for a direct comparison with Table 24. We note that the table reports CVaR results and standard deviation results even though the DRO optimization formulation does not model CVaR or standard deviation. We use these terms because they are useful summary measures to capture risk.

As the value of  $\varepsilon$  and hence risk aversion grows, the NPV drops in a manner consistent with that for the NPV-CVaR results of Table 24 except that the range here is larger; i.e., expected NPV drops from \$168.9M to \$166.6M under NPV-CVaR and from \$168.9M to \$163.0 under DRO. This larger drop in reward is coupled with a larger drop in some measures of risk. For example, the standard deviation of NPV values across the 90 scenarios drops by about \$4.1M under NPV-CVaR, dropping from about \$23M to \$18.8M, while under DRO the standard deviation drops from \$23M to \$15.8M.

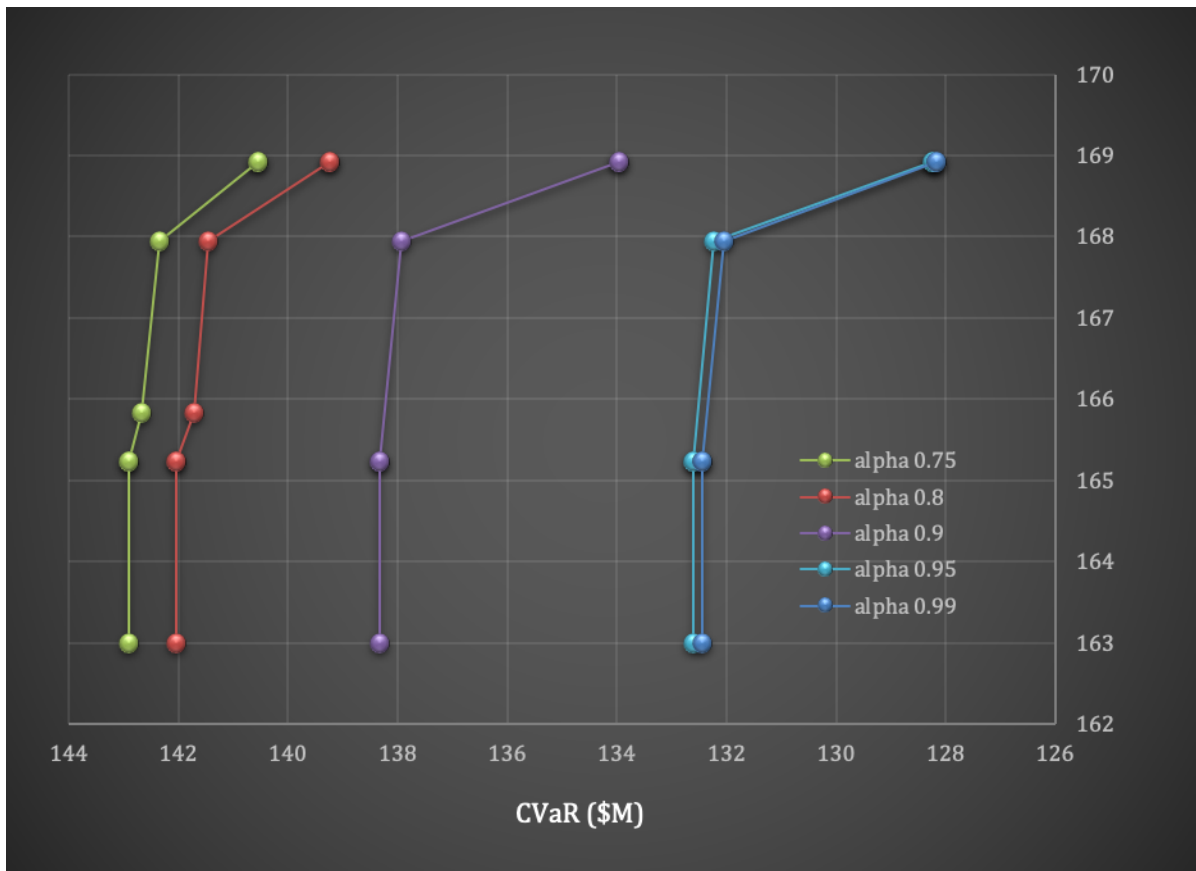
When we are risk averse, we wish to avoid low values of NPVs and hence we expect to see CVaR values grow with growing values of the radius parameters,  $\varepsilon$ . The last six rows of the table largely follow this trend. While expected NPV drops by about \$1M as we move from  $\varepsilon=0$  to  $\varepsilon=0.5$ , the  $\text{CVaR}_{0.5}$ ,  $\text{CVaR}_{0.7}$ , and  $\text{CVaR}_{0.8}$  grow by roughly \$2M. The analogous CVaR terms in the more extreme tails of the 10%, 5%, and 1% worst scenarios (i.e.,  $\text{CVaR}_{0.9}$ ,  $\text{CVaR}_{0.95}$  and  $\text{CVaR}_{0.99}$ ) grow by about \$4M. While there can be further improvements in CVaR at more extreme values of  $\varepsilon$ , such improvements are modest, less than \$0.5M. While CVaR tends to grow with  $\varepsilon$  there are exceptions, e.g., the anomalous  $\text{CVaR}_{0.75}$  term at  $\varepsilon=0.1$

amongst a trend that otherwise conforms with our expectations. There is an analogous but smaller non-monotonicity at large values of  $\varepsilon$  for  $CVaR_{0.5}$ . This decrease of about \$0.3M is not surprising because it is offset by increases of similar magnitude in CVaR at all large values of  $\alpha = 0.75, 0.80, 0.90, 0.95,$  and  $0.99$ .

**Table 25. Optimal project prioritization from solving the distributionally robust optimization model when  $\varepsilon$  ranges from 0 to 1000. This table can be compared with Table 24 when optimizing a weighted sum of NPV and CVaR.**

ID	Project name	Category	Risk	$\varepsilon=0$	$\varepsilon=0.01$	$\varepsilon=0.1$	$\varepsilon=0.5$	$\varepsilon=1$	$\varepsilon=10$	$\varepsilon=100$	$\varepsilon=1000$
1	HP feedwater heater upgrade	Optional	Med	9	9	9	8	8	8	8	8
2	Pressurizer replacement	Must do	No	4	4	13	12	12	12	12	12
3	Improvement to emergency diesel generators	Optional	Low	7	7	15	15	14	16	14	14
4	Secondary system PHM system	Optional	No	6	6	7	7	6	6	6	6
5	Replacement of two reactor coolant pumps	Must do	No	10	10	12	10	10	10	9	9
6	Seismic modification, requalification, reinforcement, improvement	Optional	Low	5	5	4	6	7	7	7	7
7	Fire protection	Must do	No	11	11	11	9	9	9	10	10
8	Service water system upgrade	Optional	Med	15	15	8	14	15	14	15	15
9	Batteries replacement	Optional	No	3	3	3	3	3	3	3	3
10	Replace CCW piping, heat exchangers, valves	Must do	No	12	12	14	13	13	11	11	11
11	Reactor vessel internals	Optional	Med	16	16	16	16	16	15	16	16
12	Reactor vessel upgrade (head included)	Must do	No	1	1	1	1	1	1	1	1
13	Replace LP turbine	Optional	Med	8	8	6	5	5	5	5	5
14	Replace instrumentation and control cables	Must do	No	2	2	2	2	2	2	2	2
15	Condenser retubing	Optional	Low	13	13	5	4	4	4	4	4
16	Replace moisture separator reheater	Optional	Med	14	14	10	11	11	13	13	13
Expected NPV (\$M)				168.92	168.92	168.59	167.95	165.84	165.23	165.23	163.00
Standard Deviation (\$M)				23.00	23.00	23.16	20.11	17.75	16.65	15.83	15.83
$CVaR_{0.5}$ (\$M)				149.83	149.83	150.50	151.82	151.96	151.61	151.61	151.61
$CVaR_{0.75}$ (\$M)				140.54	140.54	137.66	142.34	142.67	142.91	142.91	142.91
$CVaR_{0.8}$ (\$M)				139.23	139.23	136.73	141.56	141.72	142.04	142.04	142.04
$CVaR_{0.9}$ (\$M)				133.95	133.95	133.03	137.94	137.94	138.32	138.32	138.32
$CVaR_{0.95}$ (\$M)				128.23	128.24	127.30	132.23	132.23	132.61	132.61	132.61
$CVaR_{0.99}$ (\$M)				128.16	128.16	127.15	132.05	132.05	132.45	132.45	132.45

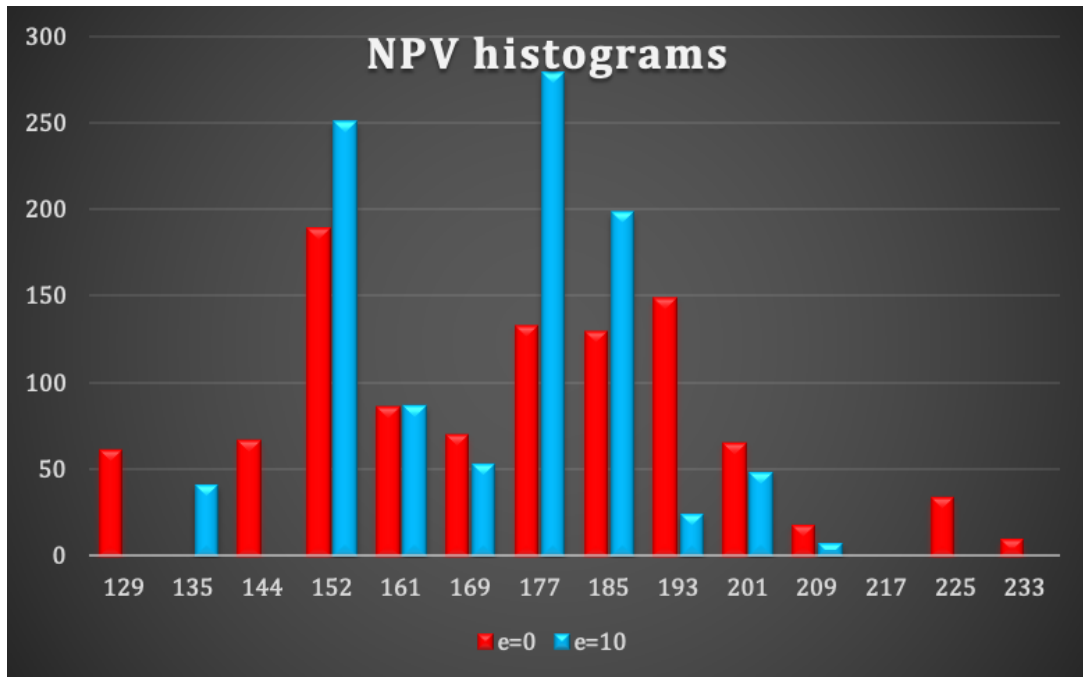
Figure 33 replicates Figure 32 except that it uses the DRO-based prioritization to compute CVaR and NPV. Broadly speaking the results under NPV-CVaR and DRO are similar: by moderately increasing the value of  $\varepsilon$  from zero we can significantly decrease risk with a modest decrement in expected NPV and the results across the two approaches are very similar. Subsequent reductions in risk require more extreme drops in NPV. We note that the DRO approach has access to a greater range of NPV values on the low end, although these yield small reductions in risk, as measured by CVaR. The corresponding reductions in the standard deviation of the NPV realizations is more substantial; see Table 25. When examining the NPV-CVaR efficient frontier, the model of Section 9.1 necessarily produces points that are not dominated in the Pareto efficient sense. Because the DRO model does not explicitly consider CVaR, it is not guaranteed to produce Pareto-efficient points, and indeed, sometimes it does not (see  $\varepsilon = 0.1$  column of Table 25). That said, it appears to do a good job of producing a relatively rich risk-return tradeoff involving risk measures CVaR and standard deviation.



**Figure 33. Pareto frontiers for different values of  $\varepsilon$  when solving the DRO model. The figure is analogous to Figure 32 for NPV-CVaR model. Note that this figure has a larger range of NPV values, but is otherwise similar to Figure 32. Note that we only include non-dominated solutions from Table 25.**

Figure 34 shows histograms of the NPV values for two epsilons,  $\varepsilon = 0$  and  $\varepsilon = 10$ . Compare the blue ( $\varepsilon = 10$ ) and red ( $\varepsilon = 0$ ) bars. It is evident that the DRO approach trims the tails and tightens the distribution. Figure 34 for DRO is the analog of Figure 31 for NPV-CVaR, and comparing the two plots

we can visually observe that the DRO approach tightens the distribution to a greater degree than does the NPV-CVaR approach.



**Figure 34. Compares histograms for two values of epsilon, 0 (red) and 10 (blue).**

Table 26 shows the frequencies with which each project is implemented under each of the plans A, B, C, and do nothing under the DRO approach for  $\varepsilon = 0$  and  $\varepsilon = 10$ . The left-half of the plot is for  $\varepsilon = 0$  and repeats the results from Table 22 in which we maximize expected NPV (breaking ties arbitrarily) while the right-half of the plot provides the risk-averse counterpart under  $\varepsilon = 10$ .

Figure 35 and Figure 36 show how nature alters probability mass function under the DRO approach. As epsilon increases, we become more risk averse and the probability mass moves to scenarios which yield lower NPV. Figure 35 shows this for the probability mass function associated with the budget scenarios and Figure 36 shows this for the medium-risk projects (low-risk projects behave similarly).

The discussion above shows the DRO formulation for the prioritization model for stochastic capital budgeting. Next we review analogous models for related problems starting with the single knapsack problem.

**Table 26. Frequencies for plans A, B, and C and do nothing under the DRO approach for  $\epsilon = 0$  and  $\epsilon = 10$ .**

ID	Project name	Category	Risk	$\epsilon = 0$					$\epsilon = 10$				
				Priority	Plan A	Plan B	Plan C	Do Nothing	Priority	Plan A	Plan B	Plan C	Do Nothing
1	HP feedwater heater upgrade	Optional	Med	9	27	63	0	0	8	41	49	0	0
2	Pressurizer replacement	Must do	No	4	90	0	0	0	12	27	63	0	0
3	Improvement to emergency diesel generators	Optional	Low	8	63	27	0	0	14	23	0	0	67
4	Secondary system PHM system	Optional	No	6	69	21	0	0	6	88	2	0	0
5	Replacement of two reactor coolant pumps	Must do	No	10	12	78	0	0	10	8	82	0	0
6	Seismic modification, requalification, reinforcement, improvement	Optional	Low	5	90	0	0	0	4	88	2	0	0
7	Fire protection	Must do	No	11	6	84	0	0	9	9	81	0	0
8	Service water system upgrade	Optional	Med	15	48	0	0	42	15	12	0	0	78
9	Batteries replacement	Optional	No	3	90	0	0	0	3	90	0	0	0
10	Replace CCW piping, heat exchangers, valves	Must do	No	12	6	71	13	0	11	7	25	58	0
11	Reactor vessel internals	Optional	Med	16	18	0	0	72	16	3	0	0	87
12	Reactor vessel upgrade (head included)	Must do	No	1	90	0	0	0	1	90	0	0	0
13	Replace LP turbine	Optional	Med	7	63	27	0	0	5	88	2	0	0
14	Replace instrumentation and control cables	Must do	No	2	90	0	0	0	2	90	0	0	0
15	Condenser retubing	Optional	Low	13	77	0	0	13	7	88	2	0	0
16	Replace moisture separator reheater	Optional	Med	14	48	5	0	37	13	27	31	0	32

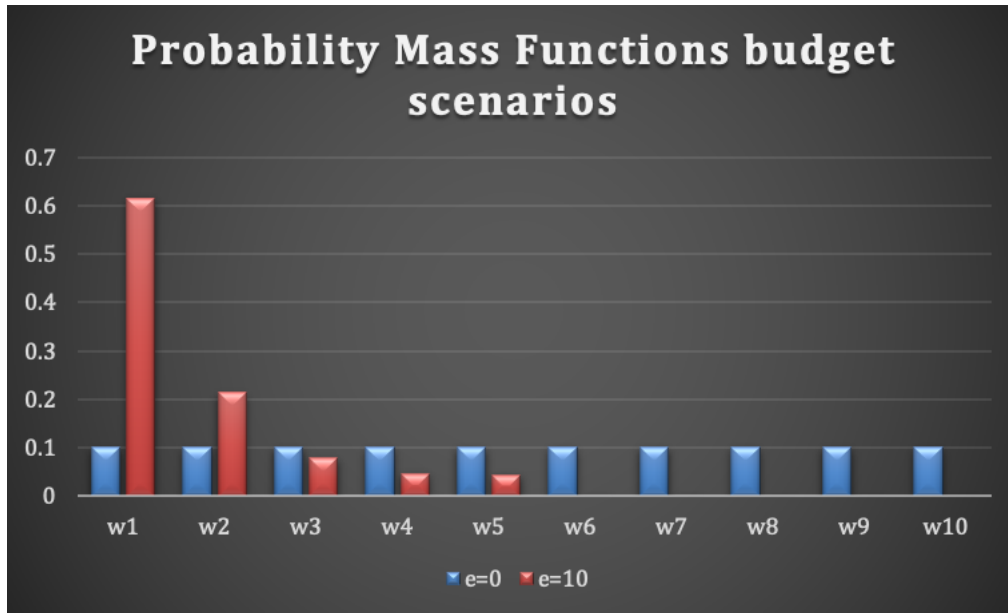


Figure 35. Probability mass functions for budget scenarios as risk aversion grows.

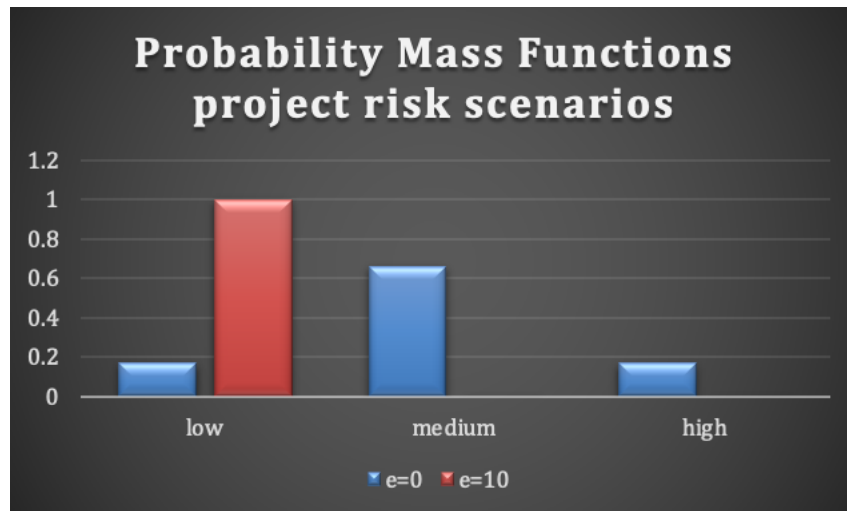


Figure 36. Probability mass functions for medium-risk project scenarios.

### 9.6.1 DRO for Single Knapsack Problem

$$\max_{x,y} \max_{\gamma,v} \left( -\gamma\varepsilon + \sum_{\sigma \in \Sigma} v^\sigma q^\sigma \right) \quad (\text{DRO6-a})$$

$$-\gamma d_{\sigma,\omega} + v^\sigma \leq \sum_{i \in N} p_i^\omega x_i^\omega \quad (\text{DRO6-b})$$

$$\sum_{i \in N} w_i^\omega x_i^\omega \leq c^\omega \quad (\text{DRO6-c})$$



$$y_{i,i'} + y_{i',i} \geq 1 \text{ and } i < i' \quad (\text{DRO6-d})$$

$$x_i^\omega \geq x_{i'}^\omega + y_{i,i'} - 1 \text{ and } i \neq i' \quad (\text{DRO6-e})$$

$$x_i^\omega, y_{ii'}^\omega \in \{0,1\}, i \neq i', i, i' \in I, \omega \in \Omega \quad (\text{DRO6-f})$$

$$\gamma \geq 0, v^\sigma \geq 0, \forall \sigma \in \Sigma \quad (\text{DRO6-g})$$

### 9.6.2 DRO for Multi-Dimensional Knapsack Problem

$$\max_{x,y} \max_{\gamma,v} \left( -\gamma \varepsilon + \sum_{\sigma \in \Sigma} v^\sigma q^\sigma \right) \quad (\text{DRO7-a})$$

$$-\gamma d_{\sigma,\omega} + v^\sigma \leq \sum_{i \in N} p_i^\omega x_i^\omega \quad (\text{DRO7-b})$$

$$\sum_{d \in D} w_{i,d}^\omega x_i^\omega \leq c_d^\omega \quad (\text{DRO7-c})$$

$$y_{i,i'} + y_{i',i} \geq 1 \text{ and } i < i' \quad (\text{DRO7-d})$$

$$x_i^\omega \geq x_{i'}^\omega + y_{i,i'} - 1 \text{ and } i \neq i' \quad (\text{DRO7-e})$$

$$x_i^\omega, y_{ii'}^\omega \in \{0,1\}, i \neq i', i, i' \in I, \omega \in \Omega \quad (\text{DRO7-f})$$

$$\gamma \geq 0, v^\sigma \geq 0, \forall \sigma \in \Sigma \quad (\text{DRO7-g})$$

### 9.6.3 DRO for Multiple Knapsack Problem

$$\max_{x,y} \max_{\gamma,v} \left( -\gamma \varepsilon + \sum_{\sigma \in \Sigma} v^\sigma q^\sigma \right) \quad (\text{DRO8-a})$$

$$-\gamma d_{\sigma,\omega} + v^\sigma \leq \sum_{m \in M} \sum_{i \in N} p_i^\omega x_{i,m}^\omega \quad (\text{DRO8-b})$$

$$\sum_{i \in N} w_i^\omega x_{i,m}^\omega \leq c_m^\omega \quad (\text{DRO8-c})$$

$$y_{i,i'} + y_{i',i} \geq 1 \text{ and } i < i' \quad (\text{DRO8-d})$$

$$\sum_{m \in M} x_{i,m}^\omega \geq \sum_{m \in M} x_{i',m}^\omega + y_{i,i'} - 1 \text{ and } i \neq i' \quad (\text{DRO8-e})$$

$$\sum_{m \in M} x_{i,m}^\omega \leq 1 \quad (\text{DRO8-f})$$

$$x_{i,m}^\omega, y_{ii'}^\omega \in \{0,1\}, i \neq i', i, i' \in I, \omega \in \Omega \quad (\text{DRO8-g})$$

$$\gamma \geq 0, v^\sigma \geq 0, \forall \sigma \in \Sigma \quad (\text{DRO8-h})$$

#### 9.6.4 DRO for Multiple-Choice Knapsack Problem

$$\max_{x,y} \max_{\gamma, \nu} \left( -\gamma \varepsilon + \sum_{\sigma \in \Sigma} \nu^\sigma q^\sigma \right) \quad (\text{DRO9-a})$$

$$-\gamma d_{\sigma, \omega} + \nu^\sigma \leq \sum_{j \in J_i} \sum_{i \in N} p_{i,j}^\omega x_{i,j}^\omega \quad (\text{DRO9-b})$$

$$\sum_{i \in N} w_{i,j}^\omega x_{i,j}^\omega \leq c^\omega \quad (\text{DRO9-c})$$

$$y_{i,i'} + y_{i',i} \geq 1 \text{ and } i < i' \quad (\text{DRO9-d})$$

$$\sum_{j=1}^{J_i-1} x_{i,j}^\omega \geq \sum_{j=1}^{J_{i'}-1} x_{i',j}^\omega + y_{i,i'} - 1 \text{ and } i \neq i' \quad (\text{DRO9-e})$$

$$\sum_{j \in J_i} x_{i,j}^\omega = 1 \quad (\text{DRO9-f})$$

$$x_{i,m}^\omega, y_{ii'}^\omega \in \{0,1\}, i \neq i', i, i' \in I, \omega \in \Omega \quad (\text{DRO9-g})$$

$$\gamma \geq 0, \nu^\sigma \geq 0, \forall \sigma \in \Sigma \quad (\text{DRO9-h})$$

## 10. LINK WITH PHM PROJECT

The use case related to development of a RIAM program has significant commonalities to the Use Case that is developing a modern, integrated, Risk-Informed Plant System Health (RI-PSH) program. Although these two use cases are similar in that they focus on plant equipment and system performance, they possess different emphases in objectives and timeframes. This is characterized in Table 27.

**Table 27. Emphases and timeframes for system health and asset management Use Cases.**

Program	Primary Timeframe	Primary Focus
PHM	Short to Intermediate Term	Engineering
RIAM	Intermediate to Long Term	Financial

RIAM applies a combination of financial and engineering evaluation methods to apply risk management technology to support plant long-term planning and investment. RIAM is intended to provide decision makers with both qualitative and quantitative information related to investments in plant assets with an objective of optimizing long-term economic value while effectively identifying and controlling enterprise risks. As described in the 2019 RI-PSH project report [2], an important set of methods and tools to support NPP long-term asset management efforts (in particular, with their application to NPPs that are anticipating operation during extended periods of time [i.e., periods of Second License Renewal – SLR]) is Integrated Life Cycle Management (ILCM) developed by Electric Power Research Institute (EPRI). The ILCM method [12] addresses the management and optimization of large capital projects for the purposes of extended plant operation. ILCM methods and accompanying software is available to EPRI member utilities; it should be noted that since all US NPP owner operators are EPRI members, ILCM is available to all operating US NPPs.

In contrast, as described in the 2019 LWRS PHM [2] report and the 2020 PHM [13] report (which is being developed in conjunction with this report), NPP ER programs are developed and implemented in accordance with the guidance provided in INPO AP-913 [14]. Additionally, regulatory requirements provided in the Maintenance Rule [18] as implemented by the industry via guidance provided in NEI 93-01 [15], focuses, to a large extent, on the reliability and availability of plant structures, systems, and components (SSCs). As a result, PHM programs have tended to focus predominantly on the engineering aspects related to ER. Additionally, the focus on items such as Maintenance Rule performance, in particular addressing performance deficiencies associated with plant SSCs classified as (a)(1), or for SSCs which possess small margins as indicated within the Mitigating Systems Performance Index (MSPI) program [16], has focused attention of RI-PSH programs on issues requiring resolution within short to intermediate timeframes. One indicator of this focus can be seen in the content of industry sponsored research to support plant ER programs. Research to address ER related issues typically is sponsored by EPRI under the Plant Engineering Program. The results of this research are used by operating NPPs around the world to support plant ER programs. To support widespread adoption of the outcomes of this research, EPRI periodically publishes a report (which is publicly available) that lists all of the products developed from this research. A review of the most recent of these reports [17] indicates a large portion of the research focuses on the engineering aspects of plant ER with short and intermediate term timeframes. It is noted that RIAM is most closely aligned with the Life Cycle Management (LCM) portion of AP-913 [14] which has a longer-term focus than the other portions of that industry guidance document.

Although the two use cases have different emphases, it is evident that they are closely related. For example, development of long-term asset management plans related to plant life extension will be dependent upon the effectiveness of the management of the health of plant SSCs achieved by the plant ER program. Conversely, anticipated financial restraints related to either current ER programs or for future investments can have an impact on decisions related to the reliability and performance of plant SSCs. As a result, as the two use cases related to PHM and RIAM progress, the LWRS collaboration is planning to continue to engage with industry stakeholders and host utilities to coordinate activities to more fully integrate the approaches to the greatest extent practicable. Some key areas where these collaborations are anticipated to occur are the following:

- (1) Evaluation of the impact of short to intermediate term investments on long-term system performance including potential impacts on plant risk (both safety and economic) and impacts on long term capital investment needs
- (2) Evaluation of the impact of long-term investment alternatives on system performance, particularly with respect to the impacts of investment limitations and deferrals on both safety and economics.

## **11. PHM-RIAM WORKFLOW**

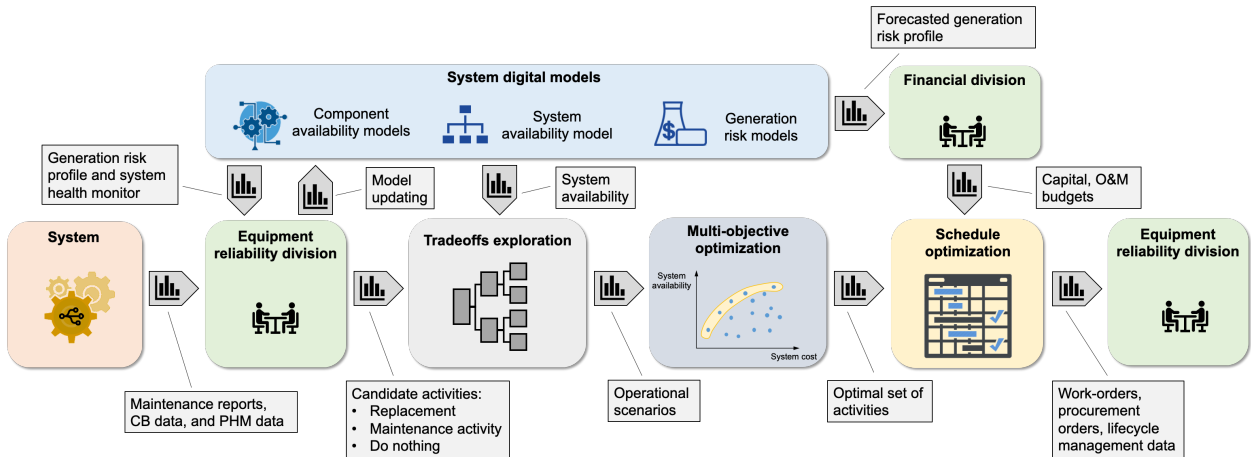
The objective of this section is to summarize how the ER decision process can be performed using the PHM-RIAM tools and methods. The ER decisions that we are targeting are:

- Which activities should be performed on the system under considerations?
- When should these activities be performed?

The process is shown in detail in Table 28 (see also Figure 37).

**Table 28. List of steps for the combined PHM-RIAM workflow.**

RISA project	Steps
PHM [13]	<ol style="list-style-type: none"> <li>1. Collect and monitor trends from ER data from the system under consideration (e.g., operation logs, issue reports, work orders, diagnostics and prognostics data)</li> <li>2. The data in Step 1 are then used to update component availability models (in terms of probability of failure or margin to failure)</li> <li>3. Information contained in component availability models (see Step 2) is then propagated to the system level to measure:               <ol style="list-style-type: none"> <li>a. System availability (in terms of probability of failure or margin to failure)</li> <li>b. Generation risk</li> </ol> </li> <li>4. Provided the information in Step 3, system engineer can provide a set of candidate activities that can be performed to either improve component availability or reduce maintenance costs (or both)</li> </ol>
RIAM	<ol style="list-style-type: none"> <li>5. Provided candidate activities listed in Step 4, it is now possible to perform a tradeoff exploration on how these candidate activities affect system availability/costs and generation risk</li> <li>6. Using the Pareto frontier analysis, it is possible to select the optimal set of component activities that guarantee system operation</li> <li>7. Based on budget (capital and/or O&amp;M) funds requirements, the optimal schedule to perform the selected activities listed in Step 6 can be developed</li> </ol>



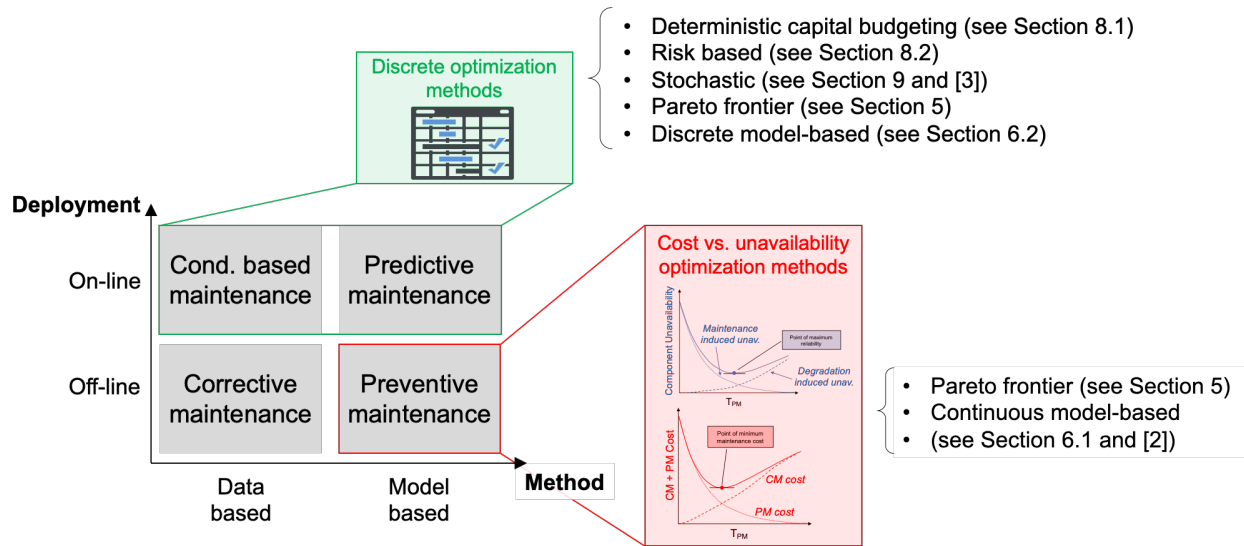
**Figure 37. From ER data to decision making using PHM-RIAM models and methods.**

## 12. LINKING MAINTENANCE APPROACHES WITH OPTIMIZATION METHODS

In [20] we have presented in detail a modeling description of the main types of maintenance approaches and how they can be included in a risk-informed analysis framework:

- Corrective maintenance
- Preventive maintenance
- Condition based maintenance
- Predictive maintenance

The approaches listed above can be classified in a 2-dimensional space based on the following characteristics: underlying analysis method (data or model based), and type of deployment (on-line or off-line). In this classification, the analysis methods can be either qualitative or quantitative. In general, the off-line methods apply qualitative approaches to classify plant SSCs, especially for decisions related to the classification of SSCs designated as RTM.



**Figure 38. Classification scheme for the considered maintenance approaches and the relative optimization methods developed in this report and in [2,3].**

As part of the RIAM project, it is now the moment to link each maintenance approach and its optimal choice of optimization method(s). Figure 38 provides a graphical form of this strategy:

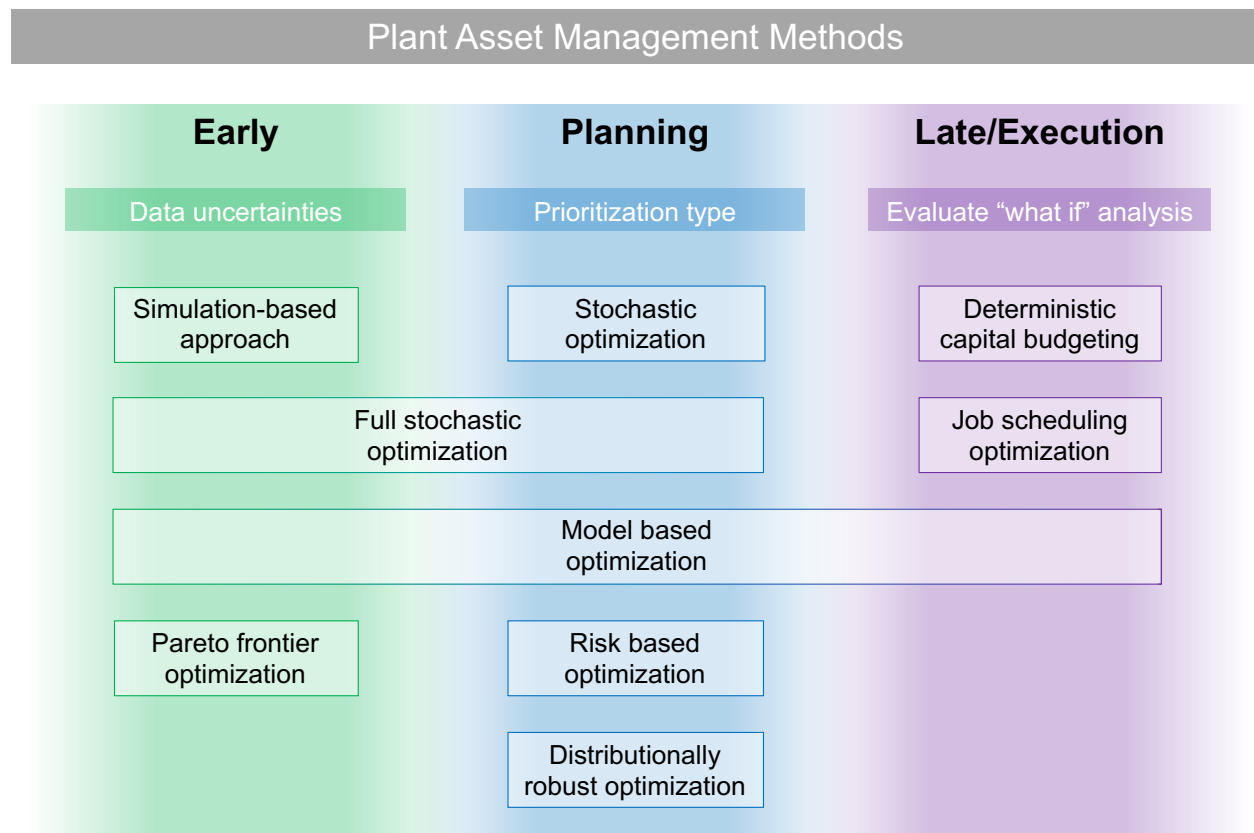
- Preventive maintenance approaches set a fixed maintenance frequency based on reliability constraints; by employing continuous optimization model-based methods it is now possible to integrate cost constraints into the determination of the optimal maintenance frequency (see Section 6.1 and [3])
- Condition based and predictive maintenance approaches employ advanced diagnostic and prognostic tools that are able to assess component status (condition based maintenance) and predict its future reliability behavior. Provided this information, it is possible to determine optimal maintenance operations (using the Pareto frontier approach shown in Section 5) and

the optimal schedule for such operations based on both economic and cost constraints (using methods shown in Sections 6.2, 8.1, 8.2, and 9).

### 13. CONCLUSIONS

In this report we have shown the recent development in the RIAM project and how they relate to its counterpart, the RI-PSH project. The RIAM project focused on the development of methods designed to optimize plant operations (e.g., maintenance and replacement schedule, optimal maintenance posture) provided system and component health and cost data.

A large variety of methods has been developed and are summarized in this report. We focused on both model-based and data-based optimization methods. The first ones are designed for very generic applications and they directly employ the models developed within the RI-PSH project. These approaches apply both continuous and discrete methods. In simple terms, these optimization methods explicitly include reliability coupled with cost models to determine optimal plant operational strategy.



**Figure 39. Overview of the developed optimization methods.**

The second class of methods target more specific use cases (e.g., project schedule optimization) and are not based on reliability models directly but they require a specific dataset that can be provided by RI-PSH analyses (e.g., component failure probability and its economic impact). This class of methods are based on the multi-dimensional knapsack problem and they aim to determine the optimal project schedule that maximize the overall NPV. In addition, several reformulations of the multi-dimensional knapsack problem have been performed during FY-20 which include a distributionally robust form and risk-measure

based approach. Depending on the available data or on the specific class of problem, the user can select the best methods based on his or her needs.

This report also presented the development of methods designed to identify optimal maintenance posture based on the Pareto frontier analysis. Rather than performing a tradeoff analysis (i.e., identify the absolute best posture), we have shown how it is possible to perform a trade space exploration approach (i.e., identify value and costs of several postures and let the analysis impose desired value and cost constraints). This is performed by identifying maintenance postures that maximize value (e.g., system availability) and minimize operational costs, i.e., the Pareto frontier in a value-cost trade space.

Figure 37 shows in a graphical form the classification of the methods being developed under the RIAM project. These methods have been classified into three different classes based on the decision stage (early stage, planning stage, or execution stage).

In FY-21 we are planning to deploy what has been developed during FY-19 and FY-20 in the RIAM and PHM project for specific use cases that are of interest to our industry partners. We will focus even more on data assimilation and integration with our developed models and methods.

## REFERENCES

- [1] R. Szilard, H. Zhang, S. Hess, J. Gaertner, D. Mandelli, S. Prescott, M. Farmer, “RISA Industry Use Case Analysis,” INL Technical Report INL/EXT-18-51012 (2018).
- [2] D. Mandelli, Z. Ma, R. Youngblood, S. St. Germain, C. Smith, P. Talbot, S. Hess, D. Dube, C. Pope, J. Miller, M. Robbins, D. Das, M. Azarian, and J. Coble, “Plant Integral Risk-informed System Health Program,” INL Technical Report INL-EXT-19-55808 (2019).
- [3] D. Mandelli, C. Wang, S. St. Germain, C. Smith, D. Morton, I. Popova, and S. Hess, “Combined Data Analytics and Risk Analysis Tool for Long Term Capital SSC Refurbishment and Replacement,” INL Technical Report INL-EXT-19-55819 (2019).
- [4] E. Delage and Y. Ye, “Distributionally Robust Optimization Under Moment Uncertainty with Application to Data-Driven Problems,” *Operations Research*, **58**, no. 3, pp. 595–612 (2010).
- [5] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution programs*, Springer Edition, 3<sup>rd</sup> edition (1996).
- [6] C.A. Mattson, A. Messac, “Pareto Frontier Based Concept Selection Under Uncertainty, with Visualization,” *Optimization and Engineering*, **6**, pp. 85–115 (2005).
- [7] A. Ramos, J. Ferreira, J. Barcelo, “Model-Based Systems Engineering: An Emerging Approach for Modern Systems,” *IEEE Transactions on Systems Man and Cybernetics*, **42**, no. 1, pp. 101-111 (2011).
- [8] A. Alfonsi, C. Rabiti, D. Mandelli, J. Cogliati, C. Wang, P. W. Talbot, D. P. Maljovec, C. Smith, “RAVEN User Guide,” INL Technical Report INL/EXT-18-44465 (2020).
- [9] R.K. Sarin, *Multi-attribute Utility Theory*, Encyclopedia of Operations Research and Management Science, Springer, Boston, MA (2013).
- [10] C. L. Smith, S.T. Wood, D. O’Neal, “Systems Analysis Programs for Hands-On Integrated Reliability Evaluations (SAPHIRE) Version 8,” NUREG/CR-7039, vol. 3 (2011).
- [11] Electric Power Research Institute (EPRI), “Computer Aided Fault Tree Analysis System (CAFTA), Version 6.0b,” EPRI Report 3002004316 (2014).
- [12] Electric Power Research Institute (EPRI), “Integrated Life Cycle Management: Status Report,” EPRI Report 1021188 (2010).
- [13] D. Mandelli, C. Wang, J. Cogliati, C. Smith, S. Hess, R. Sugrue, C. Pope, J. Miller, S. Ercanbrack, D. Cole, J. Yurko, “Integration of Data Analytics with Plant System Health Program,” INL Technical Report INL/EXT-20-59928 (2020).
- [14] Institute of Nuclear Power Operations (INPO), “AP-913 - Equipment Reliability Process Description, Revision 6,” Washington, DC (2018) (limited distribution).
- [15] Nuclear Energy Institute (NEI), “Industry Guideline for Monitoring the Effectiveness of Maintenance at Nuclear Power Plants,” NEI Report NUMARC 93-01, Rev. 4F, Washington, DC (2018).
- [16] Nuclear Energy Institute (NEI), “Regulatory Assessment Performance Indicator Guideline, Revision 7,” NEI Report 99-02 Washington, DC (2013) (<https://www.nrc.gov/docs/ML1326/ML13261A116.pdf>).
- [17] Electric Power Research Institute (EPRI), “Plant Engineering: 2019 Complete Product List”: EPRI Report 3002007859 (2020).



- [18] United States Nuclear Regulatory Commission (NRC), “10CFR 50.65 Requirements for monitoring the effectiveness of maintenance at nuclear power plants,” Washington, DC. [<https://www.nrc.gov/reading-rm/doc-collections/cfr/part050/part050-0065.html>].
- [19] D. Mandelli, C. Wang, S. St. Germain, C. Smith, D. Morton, I. Popova, and S. Hess, “Combined Data Analytics and Risk Analysis Tool for Long Term Capital SSC Refurbishment and Replacement,” INL Technical Report INL-EXT-19-55819 (2019).
- [20] R. T. Rockafellar, and S. P. Uryasev, “Optimization of Conditional Value at Risk,” *Journal of Risk*, **2**, pp. 21-42 (2000).
- [21] S. Sarykalin, G. Serraino, and S. Uryasev, “Value at Risk vs. Conditional Value at Risk in Risk Management and Optimization,” *Tutorials in Operations Research*, pp. 270-294 (2008).
- [22] G. C. Pflug, “Some Remarks on the Value at Risk and the Conditional Value At Risk,” in *Probabilistic Constrained Optimization: Methodology and Applications*, pp. 278-287, Norwell, MA (2000).
- [23] C. Acerbi, “Spectral measures of risk: A coherent representation of subjective risk aversion,” *Journal of Banking and Finance*, **26**, pp. 1505-1518 (2002).
- [24] D. Duque, and D. P. Morton, “Distributionally Robust Stochastic Dual Dynamic Programming,” Retrieved September 2020, from [http://www.optimization-online.org/DB\\_HTML/2019/12/7539.html](http://www.optimization-online.org/DB_HTML/2019/12/7539.html) (2019).
- [25] F. Hillier and G. Lieberman, *Introduction to Operations Research*, McGraw-Hill, 10th edition (2015).