Proceedings of the 2019 Winter Simulation Conference
N. Mustafee, K.-H.G. Bae, S. Lazarova-Molnar, M. Rabe, C. Szabo, P. Haas, and Y.-J. Son, eds.

COMPUTATIONAL OPERATIONS RESEARCH EXCHANGE (CORE): A CYBER-INFRASTRUCTURE FOR ANALYTICS

Yunxiao Deng
Jiajun Xu*
Carl Kesselman
Suvrajeet Sen

Epstein Department of Industrial and Systems Engineering
*Ming Hsieh Department of Electrical and Computer Engineering
University of Southern California
Olin Hall of Engineering
Los Angeles, California 90089, USA

ABSTRACT

cORe is a new cyber-infrastructure which will facilitate computational Operations Research exchange. OR models arise in many engineering domains, such as design, manufacturing, and services (e.g., banking/finance, health systems), as well as specific infrastructure-centric applications such as logistics/supply chains, power system operations, telecommunications, traffic/ transportation, and many more. In addition, modern OR tools have also been adopted in many foundational disciplines, such as computer science, machine learning, and others. Given the broad footprint of OR, the development of a robust cyber-infrastructure has the potential to not only promote greater exchange of data, models, software, and experiments but also enhance reproducibility and re-usability, both within OR, and across multiple disciplines mentioned above. cORe also has the potential to drastically reduce the computational burden on research communities which study resource allocation using analytics. This paper presents an overview of the functionality, design, and computations using cORe.

1 MOTIVATION AND REQUIREMENTS

This paper introduces a new Cyber-Infrastructure (CI) for the Operations Research (OR) community. The goals of this project are:

- Promote *reproducibility* of computational results, and enhance validation and comparison of OR models and algorithms
- Provide a *pedagogical* tool for training students.
- Promote *reusability* by sharing of data, code, methods, and evaluations across the OR community

A common thread that binds these goals together is that they are enabled only if all of the data associated with an *OR solution* (code, input data, intermediate products) are discoverable and broadly accessible in a form that could be understood and reused, with enough descriptive information to understand what the data represents, and how the Analytics process can be reproduced. Here the data includes the mathematical model (which in case of optimization is provided via special-purpose codes such a AMPL, ARENA etc.), the parameters specified for instantiating mathematical objects, and the workflow associated with the process. The outputs associated with such an experiment leads to insights, and ultimately decisions. We refer to the entire workflow as the Analytics process, and usually involves multiple paradigms, including data science, decision science, simulation, visualization etc. cORe is intended to provide a platform for *sharing OR solutions* as specified above.

It has become broadly accepted in the scientific data sharing community that by ensuring certain principles, reproducibility can be enhanced. These are the so-called FAIR principles (Madduri et al. 2018) — that it is Findable, Accessible, Interoperable, and Reusable. Specifically in the context of OR solutions, we require that the "input" is **findable** when it is identified by a unique identifier and characterized by rich metadata that describe the details of the data and Analytics workflow; **accessible** via standard protocol with access control and its metadata accessible even when the data is not; **interoperable** by using standardized terms to describe it; and **reusable** by providing accurate and relevant attributes. The cORe CI is a first step in this direction.

Sen (2006) notes that "researchers in the basic sciences (physics, chemistry, biology, etc.), and even many social scientists, not only propound new theories, but also verify how well these theories predict observations in the real world." The cORe CI will allow OR researchers to reduce the burden of experimentation with OR models and algorithms, thus allowing OR solutions to be verified/validated with greater care.

The widespread availability of data and software have now made it possible to build a CI which will allow the OR community to share OR solutions, and to enable the field to become more engaged with the scientific approach of experimentation and validation. In addition to research advances, such a CI is likely to support new applications, especially those which face a similar terrain of data and decisions. For example, many cities have begun to implement advanced OR tools for bike sharing, transportation planning, crime abatement etc. These applications may be viewed under the broad umbrella of Analytics for Smart Local Communities (ASLC). OR support of ASLC calls for a multi-faceted infrastructure which promotes an exchange of ideas through modeling and algorithmic services which integrate data resources, with appropriate models and algorithms. For example, notions such as modeling demand-response in power grids, traffic-responsive automobile routing, safety-conscious bicycle routing, and many others require an integrated system of data management, predictive and prescriptive analytics, as well as validation and visualization tools. The cORe cyber-infrastructure is designed to support such end-to-end ASLC. From a societal viewpoint, communities have a lot to gain by sharing their Analytics resources, especially since they often face similar circumstances, and often have access to similar data.

To be sure, there have been several previous forays into facilitating computational OR. Current software efforts within the OR community include COIN-OR (COmputational INfrastructure for Operations Research) for developing mathematical optimization software (Ralphs et al. (2018)), `MIPLIB-2010` (`http://miplib.zib.de/`), a library of deterministic test-instances for mixed-integer programming, `SIPLIB` (Ahmed et al. (2004)) and more recently, a simulation-optimization portal known as `simopt.org` (see Pasupathy and Henderson (2006), Pasupathy and Henderson (2011)). These resources are important, but they do not serve the role of a CI. To the best of our knowledge, the most widely used CI within the OR community has been the NEOS project, which was launched almost twenty five years ago (as a joint effort between Argonne National Labs, and Northwestern University). That project, which continues now with support from several universities (e.g., Wisconsin, ASU, and others), has been the mainstay of CI for optimization. While the services NEOS provides continues to be valuable for the optimization community, it is time to create a new CI which will incorporate many lessons of the past two decades.

The high-level goals mentioned at the outset included reproducibility and reusability. In addition, we impose the following requirements which are specific to the set of services we wish to provide. Unlike previous efforts, the focus on ASLC transcends any one genre of modeling tools, algorithms and software. Since an Analytics project typically requires coalescing data and decision sciences, the resulting workflow typically involves statistical analysis, as well as optimization algorithms. With these requirements in mind, we have designed the cORe platform to achieve the following functionality:

- Support complex multi-model learning, optimization, simulation, and other tools which have become routine in Analytics projects. Such workflow involves heterogeneous data, models, algorithms and OR resources. To be effective, cORe must serve as a repository for OR resources.

- Enable rapid implementation so that the work required for computational results can be reduced by leveraging previously reported experiments. The cORe repository of computational experiments will create an atmosphere in which alternative experimental setups will not be too onerous.
- Expedite the development of prototypes, each with its particular choice of steps in the workflow. Since an Analytics project may involve multiple prototypes, the overall task may be quite daunting. However, as users begin to share tools, ASLC projects will become less daunting. We refer to this approach to Analytics as a "crowd-sourced" CI solution, which is an extension of the spirit of the package "R" for statistics.
- By creating a search-able ecosystem of OR resources, we hope to reduce the time/effort required to carry out computational experiments. This ecosystem will also support citations via digital object identifiers (`doi`). This approach allows users to be very specific about the instance(s) being used in their test and moreover, it avoids proliferation of instances associated with the same name. For instance, the SSN (Sonet Switched Network) instance in the stochastic programming (SP) literature has at least two versions with the same name; one of them has O(10^{70}) scenarios, whereas a sampled version with 5000 scenarios is also named SSN. While the solution of the second may be a good approximation of the first, the two instances are different, and their `doi` should be different.

2 SYSTEM DESIGN

We view any Analytics application as a project network (also known as Critical Path Method (CPM) (Winston 2003)). Such networks are said to be activity-on-arc networks where each step of the (Analytics) application represents a computational service provided by a specific software. In most cases, each step/activity will represent the execution of one type of software (e.g. say a regression using "R"). Because each step requires input data, these must be specified as well. Finally as in CPM, each task can only start after all predecessor tasks have been completed. Unlike CPM however, we will not require activity durations in the current network representation, although in the future this aspect may be included when we support time-sensitive applications. The class of applications supported by the conceptual design of cORe is one that can be represented by a *directed acyclic graph*. This structure is relatively easily implemented as a sequence of steps in which all precedence must be obeyed.

Continuing the analogy with CPM, the project input description will require three specific details for any particular step (activity): Name of the Specific Step (representing the Activity), Input File Name(s)/Location(s) (representing the predecessor step), and the Output Name(s)/Location(s) (representing the successor step).

Before proceeding to the details of our platform design, we should also clarify a few additional aspirational features:

- Allow users to annotate and specify workflow (or pipelines)
- Allow users to upload papers associated with experiments, so long as no copyrights are in place
- Provide the capability to search the system for similar efforts which might have been carried out in the past. Thus a list of keywords are associated with each contribution.

For users who plan to contribute to the cORe platform, Algorithm 1 is recommended to provide a workflow for any instance. Detailed instantiation can be found in the LEO-Wyndor example in Section 4.

3 SPECIFIC DESIGN CHOICES AND RELATED WORK

The CI for cORe will leverage a system known as DERIVA (Discovery Environment for Relational Information and Versioned Assets) (Schuler et al. (2016)) which is currently used to support both small and large scientific special interest groups in the Biosciences community. The DERIVA system is a model-driven web-based interface for data discovery. Because of its roots in the experimental sciences, DERIVA is designed to support collaboration through the full life-cycle of scientific data, which is intended

Algorithm 1 recipe for workflow

```
Suppose the whole process contains $N$ steps, then
for steps $i = 1, \ldots, N$ do
    Assemble data files for step $i$, named as *Step Input File* in the platform ▷ if some files required for a step are missing, write out a warning and break
    Process data files according to methods provided in the program files for step $i$ ▷ these program files can be in Jupyter Notebook format or any files stored in github repositories
    If this step was successfully completed, then output those files which may form inputs to future steps
if all $N$ steps were successfully completed then stop
else identify which steps were not completed, and why
```

to streamline the storage and management of digital assets such as pictures, including experimental data from instruments (e.g. microscopes, sequencers and flow cytometers).

Because of the its focus on mathematical modeling, and algorithm development, the workflow for OR solutions may seem to be significantly different from the needs of life sciences research. However, the steps followed during a Biosciences experiment may involve a variety of instruments, all of which must be available for anyone else to replicate those experiments. In the case of OR, each step in the experiment requires its own "instruments" which happen to be software/algorithms which operate on certain operating systems. So long as the workflow of the OR experiment specifies these "instruments" in a specific manner, others conducting the same experiment are responsible for assembling the entire protocol for the OR experiment as well. In this sense, the experimental process is similar.

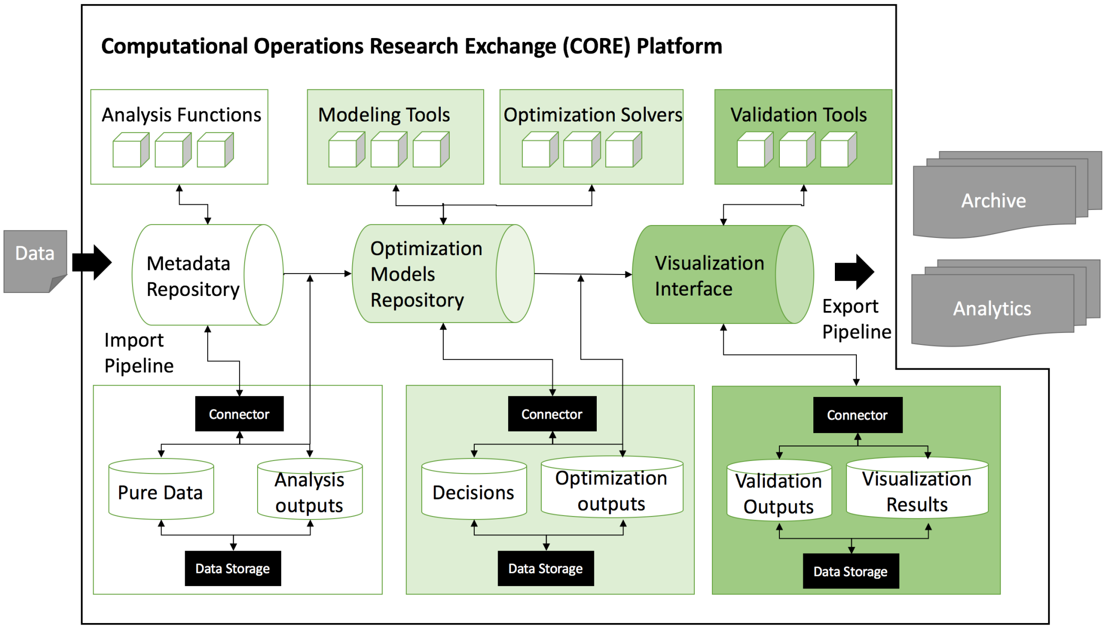
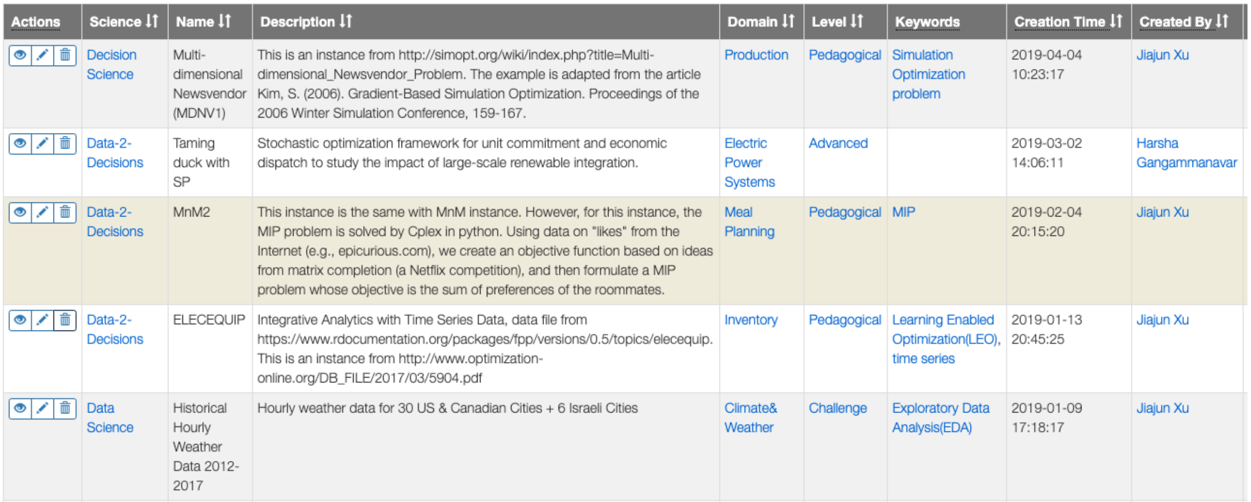


Figure 1: CORE: Computational Operations Research Exchange Platform Design

The difference between the needs of the OR community, and those of the Biosciences community is that experiments of the latter produce data which are platform-independent. Nevertheless, the protocols they follow must be specific, and anyone trying to replicate those experiments must have access to specific instruments specified by the protocol. Similarly, OR researchers will be able to download the open-source code and data, and reproduce the experiment provided they have all applicable licences installed on their platform.

In order to see how the two sets of steps require the same kind of support from DERIVA, we present a somewhat realistic example of running, the Electricity Dispatch Model for the state of Illinois (Gangammanavar et al. 2016). This experiment requires multiple steps in the workflow to represent a simulation model of wind energy production, as well as a sequential implementation of the electricity dispatch in ten minute intervals. Thus the simulated dispatch reflects plan for the next six ten-minute intervals to estimate the simulated hourly cost. Once the model, and algorithm have been developed, one begins the integration process of introducing these procedures into the larger experiment which needs to be instantiated with data for the experiment. This is accomplished through the services of DERIVA which include: a) a loosely coupled web services architecture with well defined public interfaces for every component, b) use of Entity Relationship (ER) Models that leverage standardized vocabularies, with adaptive components which can automatically respond to evolving ER models, c) model-driven user interfaces to enable navigation and discovery as the data model evolves, and d) data-oriented protocols where distributed components coordinate complex activities via data state changes. DERIVA uses the ER data model to catalog and organize assets which will be digital objects (i.e. files). Assets are characterized by contextualizing metadata which places an asset into the model by relating it to a specific entity. Additional descriptive metadata are used to describe additional attributes and relationships between assets. The components of the cORe ecosystem are shown in Figure 1. The cORe platform includes the acquisition, management, analysis and sharing of data, models and decisions, which provides the following capabilities.

- Characterization and acquisition of datasets, including input data for statistical learning and analysis, outputs from learning/optimization models, and results from validation analysis.
- Organization of data-driven decisions. The cORe system provides users with interactive ways of connecting and importing data-analysis packages/functions (such as R packages), optimization solvers and validation/visualization tools.
- The data assets will be stored in distributed in cloud based data storage systems.
- Sharing and exchange of model and data collections. The platform involves management of IP associate with data assets, which is intend to protect proprietary data and software.



Actions	Science ↓↑	Name ↓↑	Description ↓↑	Domain ↓↑	Level ↓↑	Keywords	Creation Time ↓↑	Created By ↓↑
👁 ✏ 🗑	Decision Science	Multi-dimensional Newsvendor (MDNV1)	This is an instance from http://simopt.org/wiki/index.php?title=Multi-dimensional_Newsvendor_Problem. The example is adapted from the article Kim, S. (2006). Gradient-Based Simulation Optimization. Proceedings of the 2006 Winter Simulation Conference, 159-167.	Production	Pedagogical	Simulation Optimization problem	2019-04-04 10:23:17	Jiajun Xu
👁 ✏ 🗑	Data-2-Decisions	Taming duck with SP	Stochastic optimization framework for unit commitment and economic dispatch to study the impact of large-scale renewable integration.	Electric Power Systems	Advanced		2019-03-02 14:06:11	Harsha Gangammanavar
👁 ✏ 🗑	Data-2-Decisions	MnM2	This instance is the same with MnM instance. However, for this instance, the MIP problem is solved by Cplex in python. Using data on "likes" from the Internet (e.g., epicurious.com), we create an objective function based on ideas from matrix completion (a Netflix competition), and then formulate a MIP problem whose objective is the sum of preferences of the roommates.	Meal Planning	Pedagogical	MIP	2019-02-04 20:15:20	Jiajun Xu
👁 ✏ 🗑	Data-2-Decisions	ELECEQUIP	Integrative Analytics with Time Series Data, data file from https://www.rdocumentation.org/packages/fpp/versions/0.5/topics/elecequip. This is an instance from http://www.optimization-online.org/DB_FILE/2017/03/5904.pdf	Inventory	Pedagogical	Learning Enabled Optimization(LEO), time series	2019-01-13 20:45:25	Jiajun Xu
👁 ✏ 🗑	Data Science	Historical Hourly Weather Data 2012-2017	Hourly weather data for 30 US & Canadian Cities + 6 Israeli Cities	Climate& Weather	Challenge	Exploratory Data Analysis(EDA)	2019-01-09 17:18:17	Jiajun Xu

Figure 2: Homepage of the cORe platform

4 CORE AT WORK

One of the major conveniences today is the widespread availability of local data, ranging from weather and traffic, to crime, hospitals, health-care and more. Many of these data resources are available freely on the web, while others need some level of authentication, and still others call for subscription to a service.

In the cORe platform, each experiment is classified by its scientific thrust (see Figure 2): Data Science, Decision Science and Data-2-Decision. This classification corresponds to the class of activities undertaken in an experiment. For the first category, the experiment focuses on the data science aspect, while in the second, the focus is on decision/optimization, and the third refers to the fusion of data and decision sciences.

The cORe repository classifies datasets into three categories by the level of realism: pedagogical, advanced and challenge. As the names indicate, the first category is used for educational purposes, and the other two reflect the degree of realism associated with the experiment, with advanced being less demanding than challenge instances.

To illustrate "cORe at work", we return to the three main goals summarized at the outset, namely, Reproducibility, Pedagogy, and Reusability.

- Reproducibility of computational results, and comparisons among algorithms: we provide an example from `simopt.org`, and compare results from three alternative stochastic optimization algorithms for a multi-dimensional newsvendor..
- Pedagogy. Many universities, including USC, offer a course which integrates data and decision sciences (or prescriptive and predictive Analytics). cORe supports this mission by providing a repository of examples. The one included here is simple combination of data science (matrix completion) and decision science (mixed integer programming) for meal planing, referred to as the MnM problem.
- Reusability: As mentioned earlier, publications often require editors to be able to reuse OR solutions reported in a paper. We present such an example which allows other users to run the same workflow, as described in a paper (and provided via cORe). This example, known as LEO-Wyndor, is a more advanced combination of data science (linear regression), decision science (stochastic linear programming), together with data science for model validation.

4.1 Reproducibility of Computational comparisons of algorithms: Multi-Dimensional Newsvendor

In this section we borrow a simulation optimization instance from `simopt.org`, the Multi-Dimensional Newsvendor (MDNV) Problem (Xu et al. 2018b). This instance appeared in Kim, Pasupathy, and Henderson (2015), although Harrison and Van Mieghem (1999) discussed a parametric version earlier. In this instance, a firm manufactures q products with p different resources. The resource vector $x \in \mathbb{R}^q_+$ needs to be determined before the demand vector $\xi \in \mathbb{R}^p_+$ is observed. After the demand is realized, a production vector $y \in \mathbb{R}^p_+$ is selected to maximize the profit of the firm. For given resource i and product j, let c_i be the unit investment cost for resource i, v_j be the unit profit margin for product j, a_{ij} be the amount of resource i required to produce one unit of product j. While the original instance was stated as a maximization problem, we state it below as a minimization problem:

$$\begin{aligned} \min f(x) &= c^T x + \mathbb{E}[g(x,\xi)] \\ s.t.\, 0 &\le x \le u \end{aligned} \tag{1}$$

where

$$\begin{aligned} g(x,\xi) = \min\ & -v^T y \\ s.t.\, Ay &\le x \quad \text{(capacity constraints)} \\ 0 \le y &\le \xi \quad \text{(demand constraints)} \end{aligned} \tag{2}$$

The demand vector ξ has p i.i.d. components (assumed to be non-negative), with each component having a Burr Type XII distribution, which has the cumulative distribution function:

$$F(\xi_j) = 1 - \left(1 + \xi_j^{\alpha}\right)^{-\beta}, j \in \{1, \dots, p\}$$

with parameter $\alpha = 2$, $\beta = 20$. By applying duality theory to equation 2, we have: $g(x,\xi) \geq g(x_k,\xi) + \lambda(x_k,\xi)^T(x - x_k)$, where x_k is the k-th iterate. Therefore, for a given vector ξ, dual variable $\lambda(\cdot,\xi)$ is a subgradient of $g(\cdot,\xi)$.

Several stochastic subgradient-based algorithms have been proposed for this instance by Kim et al. (2015). In the cORe platform, we compare the performance of Stochastic Approximation (SA), Robust Stochastic Approximation (RobustSA) (Nemirovski et al. 2009), and Stochastic Decomposition (SD) (Sen and Liu 2016). The data for (1), (2) are as follows: $u = 5, c^T = (5\ 5)$, $v^T = (9\ 8\ 7)$ and $A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 2 \end{pmatrix}$.

All algorithms were coded in the C language, and the experiments were conducted on a MacBook Pro with 2.7GHz Intel Core i7 processor. We run 10 replications for SA and RobustSA, with initialization point (0.1 0.1). For SA, a diminishing step size is selected, a/k, where a is 0.1 and k is iteration number. The stopping rule is either the maximum number of iteration (10000) or the following equation is satisfied:

$$\frac{a}{k}\frac{1}{N_2}|\sum_{i=1}^{N_2}\lambda(x_k,\xi_i)| < 0.0001$$

where N_2 is selected as 10. For RobustSA, the settings are follows: the total number of iteration is 10000, $N_2 = 10$, the step size is a constant equal to 0.0001. For the SD algorithm, we run 10 replications and at the end of those runs, SD produces an compromise decision based on all the replications (Sen and Liu 2016). Thus, there is only one solution and confidence interval for SD. The time reported for SD reflects all 10 replications as well as the time to obtain the compromise decision.

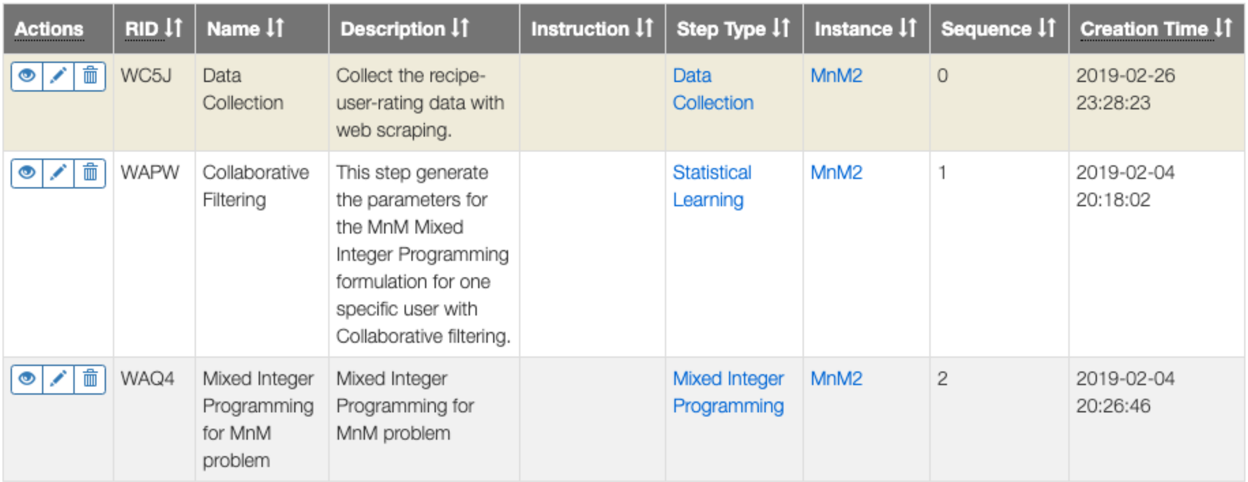
The results are reported in Table 4.1. For the notation in the table: *Rep.* is the number of replication, *Sol.* is the optimal solution, *CI* is the 95% confidence interval for the objective function and *Time* is the CPU time in seconds.

Table 4.1 *SA vs. RobustSA vs. SD on the MDNV problem*

Rep	SA			RobustSA			SD (10 Rep)		
	Sol.	CI	Time	Sol.	CI	Time	Sol	CI	Time
1	[0.1735,0.2293]	[-0.8479,-0.8395]	8.83	[0.1746,0.2270]	[-0.8479,-0.8395]	19.25	[0.1698,0.2279]	[-0.8479,-0.8394]	12.86
2	[0.1717,0.2286]	[-0.8481,-0.8396]	8.87	[0.1735,0.2290]	[-0.8479,-0.8394]	19.17			
3	[0.1742,0.2293]	[-0.8479,-0.8395]	8.41	[0.1727,0.2273]	[-0.8480,-0.8396]	19.34			
4	[0.1733,0.2289]	[-0.8480,-0.8395]	7.43	[0.1734,0.2296]	[-0.8479,-0.8394]	19.29			
5	[0.1723,0.2276]	[-0.8481,-0.8396]	9.38	[0.1714,0.2290]	[-0.8482,-0.8397]	18.99			
6	[0.1730,0.2283]	[-0.8479,-0.8395]	8.64	[0.1730,0.2289]	[-0.8480,-0.8395]	19.37			
7	[0.1725,0.2284]	[-0.8481,-0.8396]	6.48	[0.1715,0.2285]	[-0.8482,-0.8398]	20.19			
8	[0.1742,0.2284]	[-0.8479,-0.8394]	8.77	[0.1752,0.2304]	[-0.8475,-0.8391]	19.31			
9	[0.1726,0.2290]	[-0.8479,-0.8394]	8.36	[0.1705,0.2275]	[-0.8481,-0.8396]	20.04			
10	[0.1730,0.2302]	[-0.8479,-0.8394]	7.99	[0.1734,0.2283]	[-0.8480,-0.8395]	19.64			

4.2 Pedagogical example: Meal Planning for the New Millennium

We refer to this pedagogical example as the Meal Planning for the New Millennium (MnM2) Problem (Xu et al. 2018a). This problem is inspired by the famous Diet Problem (Garille and Gass 2001). Unlike the traditional diet problem, the meal planning problem requires the choice of recipes, and therefore leads to a model with binary decisions on whether a recipes is chosen or not for a given week. In addition, the meal planning problem is not based on minimizing the cost of diet; instead, this new problem gives the users a choice to include their taste as an objective. The preference rating is based on the feedback for recipes available online. In this particular instance, we first use web scraping techniques to gather the rating data and nutrition data from a recipe website. Later, a collaborative filtering method (matrix



Actions	RID	Name	Description	Instruction	Step Type	Instance	Sequence	Creation Time
👁 ✎ 🗑	WC5J	Data Collection	Collect the recipe-user-rating data with web scraping.		Data Collection	MnM2	0	2019-02-26 23:28:23
👁 ✎ 🗑	WAPW	Collaborative Filtering	This step generate the parameters for the MnM Mixed Integer Programming formulation for one specific user with Collaborative filtering.		Statistical Learning	MnM2	1	2019-02-04 20:18:02
👁 ✎ 🗑	WAQ4	Mixed Integer Programming for MnM problem	Mixed Integer Programming for MnM problem		Mixed Integer Programming	MnM2	2	2019-02-04 20:26:46

Figure 3: MnM2 problem on cORe platform

completion) is applied to predict the recipe preferences for an individual. As for the optimization side, the goal is to choose the most highly rated meal plan under nutrition, budgetary and scheduling constraints. In our setting, two roommates are considered to cook, share, and pay for meals over five days each week. Thus, the solution should satisfy not only the nutrition requirements but also schedule constraints. In the cORe platform, this MnM2 problem is classified as a pedagogical example.

Figure 3 shows the main page of the MnM2 problem in our cORe platform. The MnM2 instance consists of three phases: Data collection (web scraping), Statistical Learning (matrix completion), and Mixed Integer Programming (optimal meal plan). On the left hand side, the user can click the buttons to view or edit each phase. We encourage the reader to examine the Jupyter Notebook file to get an idea of the details necessary to undertake these steps.

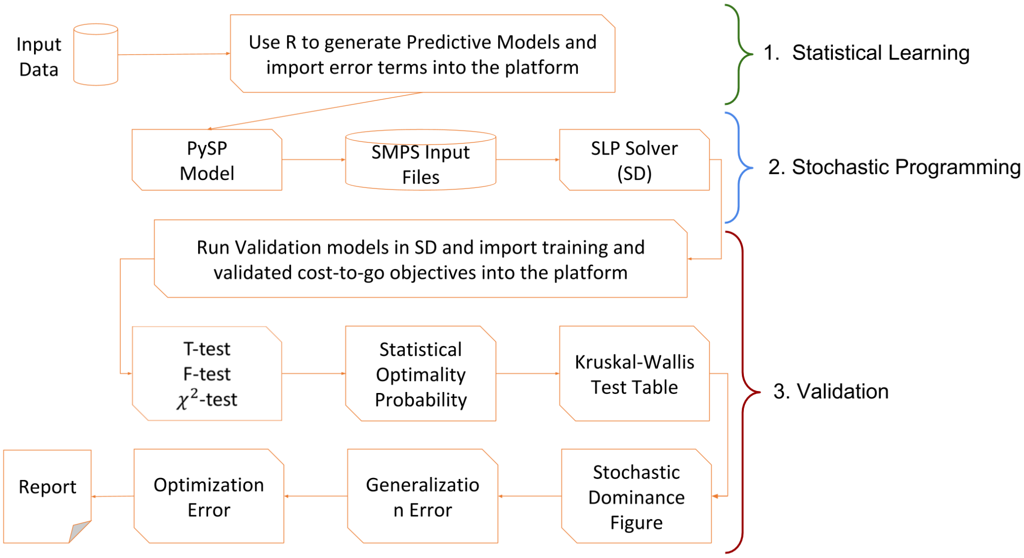

Input Data
Use R to generate Predictive Models and import error terms into the platform
1. Statistical Learning
PySP Model
SMPS Input Files
SLP Solver (SD)
2. Stochastic Programming
Run Validation models in SD and import training and validated cost-to-go objectives into the platform
T-test
F-test
χ^2-test
Statistical Optimality Probability
Kruskal-Wallis Test Table
3. Validation
Report
Optimization Error
Generalizatio n Error
Stochastic Dominance Figure

Figure 4: Workflow of LEO-Wyndor in cORe Platform

4.3 Reusability of a research example: LEO-Wyndor

Finally, we present an example where new research known as Learning Enabled Optimization (LEO) is introduced via a pedagogical problem known as Wyndor, borrowed from Hillier and Lieberman (1995). Our example, LEO-Wyndor Problem (Deng et al. 2018), extends the original Wyndor model to one in which the production plan is to be made while bearing in mind that the allocation of the advertising effort (time slots on TV or radio) affects sales of different types of products, while the original Wyndor model was intended to use product demand as input, and the production plan was intended to maximize profit under production constraints, under the assumption that the company could sell any quantity of products produced. In other words, production capacities were the main constraints. For this example, a statistical model, Multivariate Linear Regression (MLR) model, is used to predict future sales (W), considering the advertising slots (Z) on TV and radio. In this example, the advertising decisions constitute a bet on the first stage (advertising) decisions x, and the second stage decisions are the production planning choices, given firm orders (sales).

The whole process for this model includes three phases: Statistical Learning, Stochastic Programming and Validation, which are listed in the page of LEO-Wyndor instance of the cORe platform. The workflow for this instance is included in Figure 4. Multiple models have been implemented for comparison, including deterministic forecasts (DF), normally distributed and un/correlated (NDU/NDC) demands and empirical additive errors (EAE) models. A specific numerical instance of this model can be found on the cORe platform (https://core.isrd.isi.edu).

5 CONCLUSIONS

This paper has presented a new cyber-infrastructure for the OR community, and is intended to help researchers share OR solutions which includes data, models, codes, and experiments which can be reproduced without as much programming as is necessary today. We are hopeful that the OR community will adopt this framework for their research.

ACKNOWLEDGMENTS

This work was supported by grant 1822327 from the National Science Foundation.

REFERENCES

Ahmed, S., R. Garcia, N. Kong, L. Ntaimo, G. Parija, F. Qiu, and S. Sen. 2004. "SIPLIB: A stochastic integer programming test problem library". http://www2.isye.gatech.edu/~sahmed/siplib, accessed 28th June 2019.

Deng, Y., J. Xu, and S. Sen. 2018. "LEO-Wyndor Problem". https://doi.org/10.25551/W1QR.

Gangammanavar, H., S. Sen, and V. M. Zavala. 2016. "Stochastic optimization of sub-hourly economic dispatch with wind energy". *IEEE Transactions on Power Systems* 31(2):949–959.

Garille, S. G., and S. I. Gass. 2001. "Stigler's diet problem revisited". *Operations Research* 49(1):1–13.

Harrison, J. M., and J. A. Van Mieghem. 1999. "Multi-resource investment strategies: Operational hedging under demand uncertainty". *European Journal of Operational Research* 113(1):17–29.

Hillier, F. S., and G. J. Lieberman. 1995. *Introduction to operations research*. McGraw-Hill Science, Engineering Mathematics.

Kim, S., R. Pasupathy, and S. G. Henderson. 2015. *A Guide to Sample Average Approximation*, 207–243. New York, NY: Springer New York.

Madduri, R. K., K. Chard, M. D'arcy, S. C. Jung, A. Rodriguez, D. Sulakhe, E. W. Deutsch, C. Funk, B. Heavner, M. Richards et al. 2018. "Reproducible big data science: A case study in continuous FAIRness". *BioRxiv*:268755.

Nemirovski, A., A. Juditsky, G. Lan, and A. Shapiro. 2009. "Robust stochastic approximation approach to stochastic programming". *SIAM Journal on optimization* 19(4):1574–1609.

Pasupathy, R., and S. G. Henderson. 2006. "A testbed of simulation-optimization problems". In *Proceedings of the 2006 winter simulation conference*, 255–263. IEEE.

Pasupathy, R., and S. G. Henderson. 2011. "SimOpt: A library of simulation optimization problems". In *Proceedings of the Winter Simulation Conference*, 4080–4090. Winter Simulation Conference.

Ralphs, T.K., and Vigerske, S. and Waechter, A. 2018. "COIN-OR Build Tools 0.8". https://github.com/coin-or-tools/BuildTools/, accessed 28th June 2019.

Schuler, R. E., C. Kesselman, and K. Czajkowski. 2016. "Accelerating data-driven discovery with scientific asset management". In *e-Science (e-Science), 2016 IEEE 12th International Conference on*, 31–40. IEEE.

Sen, S. 2006. "On bridging the gap between academe and industry in OR/MS". *OR/MS Today (The INFORMS Professional Magazine)*:31–33.

Sen, S., and Y. Liu. 2016. "Mitigating uncertainty via compromise decisions in two-stage stochastic linear programming: Variance reduction". *Operations Research* 64(6):1422–1437.

Winston, W. L. 2003. *Introduction to Mathematical Programming: Applications and Algorithms*. Duxbury Resource Center.

Xu, J., Y. Deng, and S. Sen. 2018a. "Meal Planning for the New Millennium". https://doi.org/10.25551/WAPT.

Xu, J., Y. Deng, and S. Sen. 2018b. "Multi-Dimensional Newsvendor (MDNV) Problem". https://doi.org/10.25551/WCF8.

AUTHOR BIOGRAPHIES

Yunxiao Deng is a software engineer at Google, who has been working on traffic load tuning and optimization for Search infrastructure. She graduated from the University of Southern California with a Ph.D. in operations research. Her research interests are in stochastic optimization and statistical learning, particularly decomposition algorithms for applications of predictive stochastic programming. Her email address is yunxiaod@usc.edu

Jiajun Xu is a third year PhD student in Ming Hsieh Department of Electrical and Computer Engineering at the University of Southern California. His current research interests include Stochastic Programming, Mixed Integer Programming and their applications. He received his B.S. degree in Applied Physics from the University of Science and Technology of China (USTC) in 2016. His email address is jiajunx@usc.edu

Carl Kesselman is a Dean's Professor in the Epstein Department of Industrial and Systems Engineering, and the Director of the Informatics Systems Research Division in the Informations Sciences Institute and the Center for Excellence in Discovery Informatics ant the Michelson Center for Convergent Biosciences all at the University of Southern California. He receved a PhD in Computer Science from the University of California at Los Angeles. His research interests are in large scale information systems, scientific reproducibility and distributed data management systems. He is Fellow in the Association for Computing Machenery and the British Computing Society. His email address is carl@isi.edu

Suvrajeet Sen is Professor at the Daniel J. Epstein Department of Industrial and Systems Engineering at the University of Southern California. Prior to joining USC, he was a Professor at Ohio State University and University of Arizona. He has also served as the Program Director of OR as well as Service Enterprise Systems at the National Science Foundation. Professor Sen's research is devoted to many categories of optimization models, and he has published over a hundred papers, with the vast majority of them dealing with models, algorithms and applications of Stochastic Programming problems. In 2015, this research and his groups contributions were recognized by the INFORMS Computing Society for seminal work on Stochastic Mixed-Integer Programming. Professor Sen was instrumental in founding the INFORMS Optimization Society in 1995, and has also served as its Chair (2015-16). He is a Fellow of INFORMS and has served on several editorial boards. His email address is s.sen@usc.edu