

DDCA Summer 2019 Report

Gwendolyn J. Chee, Roberto E. Fairhurst, Kathryn D. Huff

July 8, 2019

1 Introduction

Nuclear Fuel Cycle (NFC) simulation scenarios are constrained objective functions. The objectives are systemic demands such as "1% power growth", while an example of a constraint is the availability of new nuclear technology. To aid in the setup of nuclear fuel cycle simulations, NFC simulators should bring demand responsive deployment decisions into the dynamics of the simulation logic [1]. While automated power production deployment is common in most fuel cycle simulators, automated deployment of supportive fuel cycle facilities is non-existent.

Instead, the user must detail the deployment timeline of all supporting facilities or have infinite capacity support facilities. Thus, a next generation NFC simulator should predictively and automatically deploy fuel cycle facilities to meet user defined power demand.

CYCLUS is an agent-based nuclear fuel cycle simulation framework [2]. Each entity (i.e. Region, Institution, or Facility) in the fuel cycle is modeled as an agent. Institution agents are responsible for deploying and decommissioning facility agents and can represent a legal operating organization such as a utility, government, etc [2].

The Demand-Driven CYCAMORE Archetypes project (NEUP-FY16-10512) aims to develop CYCLUS's demand-driven deployment capabilities. This capability is developed in the form of a CYCLUS Institution agent that deploys facilities to meet the front-end and back-end fuel cycle demands based on a user-defined commodity demand. Its goal is to meet supply for any commodity while minimizing undersupply. This demand-driven deployment capability is referred to as `d3ploy`.

In this paper, we will explain the capabilities of `d3ploy` and demonstrate how `d3ploy` is used to meet the primary objective of minimizing undersupply of all commodities in a simulation. The goal is to study a basic transition scenarios with constant, linearly increasing and sinusoidal power demand. Such a study provides recommendations and insights to inform decisions about parameter inputs when setting up larger transition scenarios that include many facilities. The last cases analyzed are such transition scenarios.

2 D3ploy capabilities

2.1 Core Capability of `d3ploy`

At each time step, `d3ploy` predicts demand and supply of each commodity for the next time step. Then, `d3ploy` deploys facilities to meet predicted

demand. `D3ploy`'s primary objective is to minimize the number of time steps of undersupply of any commodity.

When there is a predicted undersupply of a commodity, `d3ploy` looks at what facilities it has that provides that commodity and will deploy the fewest number of facilities to meet the predicted demand. This logic is available in `solver.py`.

2.2 Basic User-Defined Input Variables

The user is able to input specific variables to customize their simulation. Descriptions of each input variable can be found in the README of the `d3ploy` github repository.

Essentially, the user must define the facilities for the institution to control and their corresponding capacities. The user must also define the driving commodity, its demand equation and what calculation method the institution predicts demand and supply with.

Users also have the option to give a time dependent equation that governs preference for that facility compared to other facilities that provide the same commodity. The user also has an option to constrain deployment of a facility until there is an accumulation of the inventory of a specific commodity. The user can also define an initial list of facilities that are present in the institution at the beginning of the simulation.

2.3 Prediction Algorithms

Three interchangeable algorithm types govern demand and supply predictions: non-optimizing (NO), deterministic optimizing (DO), and stochastic optimizing (SO).

There are three methods implemented for the non-optimizing model: Moving Average (MA), autoregressive moving average (ARMA), and autoregressive conditional heteroskedasticity (ARCH). There are four methods implemented for the deterministic optimizing model: Polynomial fit regression (POLY), simple exponential smoothing (EXP_SMOOTHING), triple exponential smoothing (HOLT_WINTERS) and fast fourier transform (FFT). There is one method implemented for stochastic optimizing model: stepwise seasonal (SW_SEASONAL).

The user can choose which prediction algorithm governs each specific `d3ploy` commodity. The effectiveness of a prediction algorithm depends on the type of power demand in a scenario and the type of commodity. For example, the triple exponential smoothing method is most effective for

predicting demand and supply for the power commodity in a scenario with a sinusoidal power demand as opposed to a linearly increasing power demand. Likewise, the fast fourier transform method is more effective than triple exponential smoothing for the non-power commodities in the same scenario.

2.4 Difference between Demand and Supply Driven Institutions

Within `d3ploy`, there are two institutions: `DemandDrivenDeploymentInst` and `SupplyDrivenDeploymentInst`. The prior is used for the front-end of the fuel cycle and the latter is used for the back-end. Front-end facilities are those that exist before the reactor in a nuclear fuel cycle, such as a fuel fabrication facility. Back-end facilities go after the reactor in a nuclear fuel cycle; one example would be a reprocessing facility. The reason for this separation is to let facilities have the choice to demand for supply or demand for capacity. For example, in the front-end facilities, the reactor has a demand for fuel that triggers the deployment of fuel fabrication facilities. Such facilities will create a supply to meet the demand. Whereas, for the back end facilities, the reactor generates spent fuel, there is a demand for a waste repository facility to accept the spent fuel. This triggers the deployment of a waste repository that will create a capacity to receive the available supply of spent fuel.

2.5 Installed Capacity

The user can choose between two distinct methods for deploying facilities: based on the difference between predicted demand and predicted supply, or the difference between predicted demand and installed capacity. There are two main reasons for wanting to use installed capacity over predicted supply. The first is for facilities that provide intermittent supply, such as a reactor facility that has a designated refueling time. During time steps where a reactor is refueling, the user might not want `d3ploy` to deploy more facilities to make up for the lack of supply caused by this one time step gap in supply. The second is for situations where the input commodity for a facility has run out in a simulation, and the facility that produces the input commodity is no longer commissionable. Therefore, with the demand for the output commodity of that facility, `d3ploy` would deploy that facility in an attempt to meet the demand, however due to the lack of available input commodity the facility would be unable to supply its output commodity. For example, in a transition scenario to fast reactors that require plutonium

from Light Water Reactor (LWR)'s spent nuclear fuel (SNF), if the fast reactors' demand for plutonium exceeds the inventory provided by LWRs prior to decommissioning, `d3ploy` will deploy mixer facilities in an effort to produce fast reactor fuel despite the lack of plutonium with which to make the fuel. This is an example of a poorly set up transition scenario.

2.6 Supply/Capacity Buffer

In `DemandDrivenDeploymentInst`, the user can choose to provide a buffer for the predicted supply; doing so will cause `d3ploy` to ensure that predicted supply meets the predicted demand plus the additional buffer.

Similarly, in `SupplyDrivenDeploymentInst`, the user can choose to provide a buffer for the predicted capacity so that `d3ploy` will ensure predicted capacity meets the predicted supply with the additional buffer. These buffers can be defined as either a percentage value or an absolute value.

3 Demonstration of `d3ploy` capabilities

To demonstrate `d3ploy`'s capabilities we run simulations with constant, linearly increasing, and sinusoidal power demand. A balance between the various system parameters must be met for each type of simulation to minimize the undersupply and under capacity for the various commodities.

These simulations were basic transition scenarios that only included three types of facilities: `source`, `reactor` and `sink`. All of the simulations began with ten reactor facilities, `reactor1` to `reactor10`. These reactors had staggered cycle lengths and lifetimes so that they did not perform refuelings or undergo decommissioning simultaneously. `D3ploy` deploys reactor facilities of the `new reactor` type to correct the power undersupply that occurred when the ten initial reactor facilities began to decommission.

All the simulations deployed facilities based on the relationship between predicted demand and installed capacity, a capability discussed in the previous section. Table 1 shows the simulation parameters that are consistent across all the discussed scenarios. Table 2 displays the number of time steps where there was an undersupply for each commodity.

The reason for setting up these basic transition scenarios is to demonstrate `d3ploy`'s capabilities for use in simulating transition scenarios and also to inform decisions about parameter inputs when setting up larger demand transition scenarios that include many facilities.

Table 1: Transition Scenario Parameters for the constant, linear increasing, and sinusoidal power demand simulations.

Parameters	Description
Facilities Present	Source (Capacity: 3000kg), Reactor (Capacity: 1000MW), Sink (Capacity: 50000kg)
New Reactor Parameters	Cycle time: 18, Refuel time: 1
Driving Commodity	Power

Table 2: Undersupply results for each commodity in each scenario.

Transition Scenario	Commodity	No. of time steps with undersupply
Constant Power	Fuel	1
	Power	0
	Spent Fuel	0
Linearly Increasing Power	Fuel	1
	Power	0
	Spent Fuel	0
Sinusoidal Power	Fuel	1
	Power	1
	Spent Fuel	0

3.1 Transition Scenario: Constant Demand

This section shows a constant power transition scenario. Table 3 displays the simulation parameters. The input file used to generate this simulation can be found in:

/d3ploy/input/constant_transition.xml

and the file used to run the simulation and generate the plots can be found in:

/d3ploy/tests/performance_tests/algorithm_performance_tests_transitions.py

Figures 1a, 1b and 1c demonstrate the capability of `d3ploy` to deploy reactors and supporting facilities to meet the user determined power demand and subsequently demanded secondary commodities with the minimal number of time steps that result in an undersupply.

Table 2 shows the number of time steps where there was an undersupply

Table 3: Constant Power Demand Transition Scenario’s Parameters.

	Parameters	Description
Overall	Demand Equation	10000 MW
Power Commodity	Prediction Method	Fast Fourier Transform
	Supply Buffer	3000 MW
Fuel Commodity	Prediction Method	Moving Average
	Supply Buffer	0 kg
Spent Fuel Commodity	Prediction Method	Moving Average
	Capacity Buffer	0 kg

for each commodity in this scenario. In figure 1a, there are no time steps where the supply of power falls under demand.

The use of the fast fourier transform method for predicting the demand and setting the supply buffer to 3000MW (the capacity of 3 reactors) minimized the number of undersupply time steps.

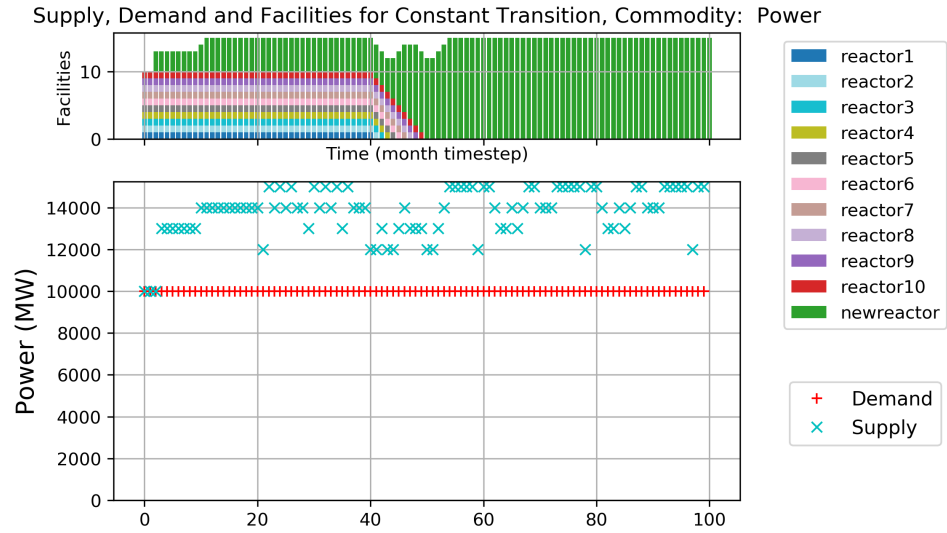
It is important to perform a small sensitivity analysis of the size of buffer used for each commodity to ensure that there is no undersupply based on the nuances of any given facility type: refueling in a reactor, etc..

In figure 1b, a facility with a large throughput of fuel is initially deployed to meet the large initial fuel demand from the startup of ten reactors. This is a reflection of reality, wherein reactor operators will accumulate an appropriate amount of fuel inventory before bringing reactors online. There is one time step where there is an undersupply following the decommissioning of the large initial fuel production facility. This is unavoidable, as the prediction methods harnessed by `d3ploy` are unable to foresee this sudden drop in demand.

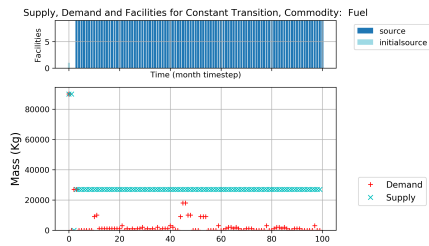
For simulations such as this, where a facility requires a large initial amount of some commodity, the user should add an initial facility with a large production capacity that exists for only the first few time steps in the simulation; this prevents `d3ploy` from deploying a large number of supporting facilities that end up being redundant later in the simulation. Alternatively, this could be circumvented by introducing decommissioning capability into `d3ploy`.

3.2 Transition Scenario: Linearly Increasing Demand

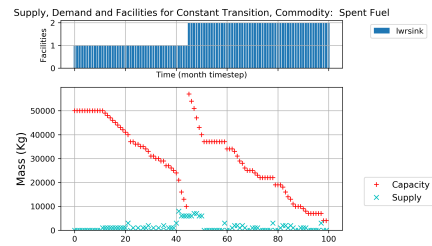
This section presents a transition scenario with a linearly increasing power demand. Table 4 displays the simulation parameters used in this transition scenario.



(a) Power demand and supply plot.



(b) Fuel demand and supply plot.



(c) Spent Fuel demand and supply plot.

Figure 1: Transition Scenario: Constant Power Demand of 10000MW.

Table 4: Linearly Increasing Power Demand Transition Scenario’s Parameters.

	Parameters	Description
Overall	Demand Equation	Time<40: 10000 MW, Time>40: 250*t MW
Power Commodity	Prediction Method	Fast Fourier Transform
	Supply Buffer	2000 MW
Fuel Commodity	Prediction Method	Moving Average
	Supply Buffer	1000 kg
Spent Fuel Commodity	Prediction Method	Fast Fourier Transform
	Capacity Buffer	0 kg

Figures 2a, 2b and 2c demonstrate the capability of `d3ploy` to deploy reactors and supporting facilities to meet the user determined power demand and subsequently required secondary commodities for a linearly increasing power demand.

This scenario made use of the fast fourier transform method for predicting power demand, similar to what the constant power demand transition scenario used. The power supply buffer in this case was an absolute value of 2000MW.

The input file used to generate this simulation can be found in:

`/d3ploy/input/growing_transition.xml`

and the file used to run the simulation and generate the plots can be found in:

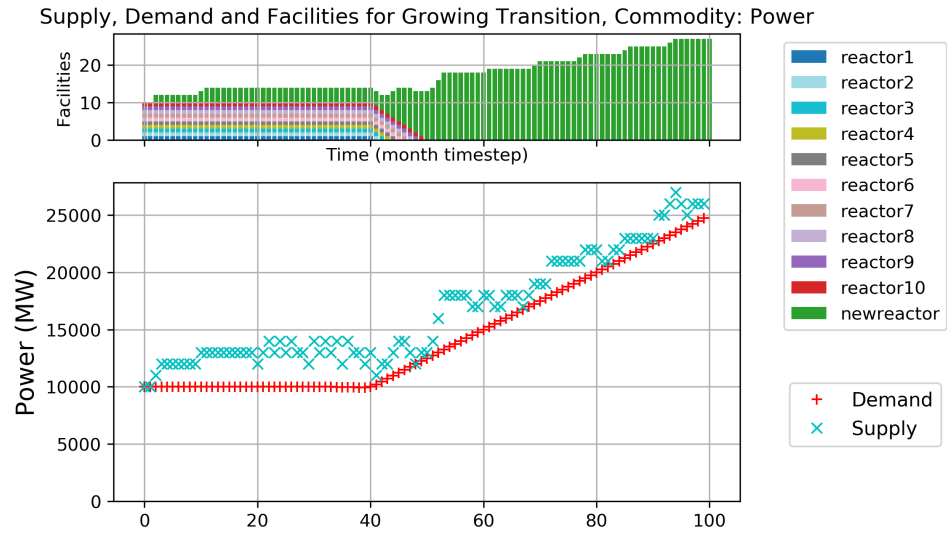
`/d3ploy/tests/performance_tests/algorithm_performance_tests_transitions.py`

3.3 Transition Scenario: Sinusoidal Demand

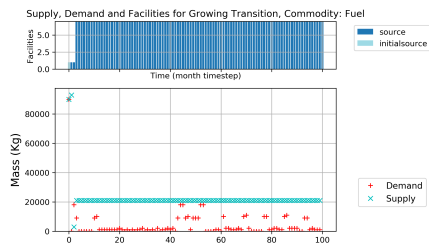
This section shows a transition scenario with sinusoidal power demand. A sinusoidal power demand is a more accurate reflection of the real world where consumption is higher in the winter and summer and lower in the spring and fall. Table 5 displays the simulation parameters used in this transition scenario. The power demand had an amplitude of 1000MW.

Figures 3a, 3b and 3c demonstrate the capability of `d3ploy` to deploy reactors and supporting facilities to meet the user determined power demand and subsequently required secondary commodities for a sinusoidal power demand.

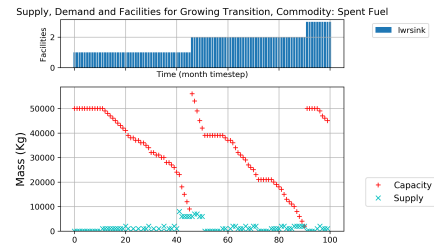
For a sinusoidal power demand, the use of the triple exponential method (Holt-Winters) for predicting demand is more effective than the fast fourier



(a) Power demand and supply plot.



(b) Fuel demand and supply plot.



(c) Spent Fuel demand and supply plot.

Figure 2: Transition Scenario: Linearly Increasing Power Demand.

transform method previously used in the constant and linearly increasing power demand transition scenarios. This is because the triple exponential smoothing method excels in forecasting data points for repetitive seasonal series of data.

Table 5: Sinusoidal Power Demand Transition Scenario’s Parameters.

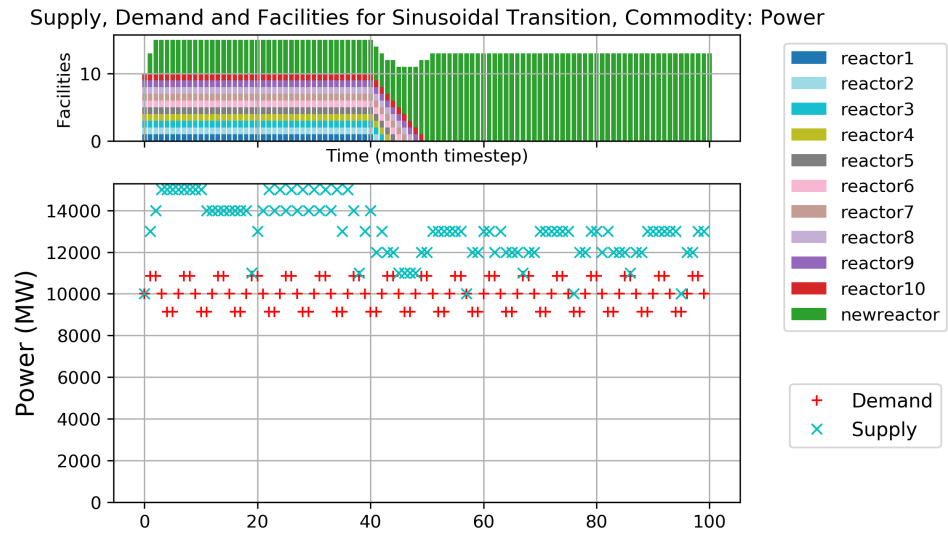
	Parameters	Description
Overall	Demand Equation	$1000\sin(\frac{\pi*t}{3}) + 10000$
Power Commodity	Prediction Method	Triple Exponential Smoothing
	Supply Buffer	2000 MW
Fuel Commodity	Prediction Method	Moving Average
	Supply Buffer	1000 kg
Spent Fuel Commodity	Prediction Method	Fast Fourier Transform
	Capacity Buffer	0 kg

The input file used to generate this simulation can be found in:

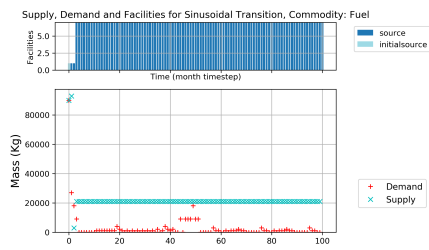
/d3ploy/input/sine_transition.xml

and the file used to run the simulation and generate the plots can be found in:

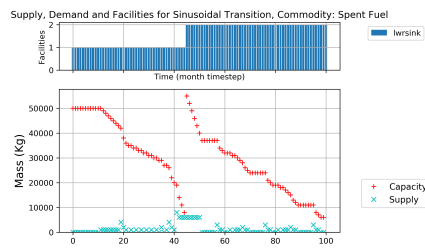
/d3ploy/tests/performance_tests/algorithm_performance_tests_transitions.py



(a) Power demand and supply plot.



(b) Fuel demand and supply plot.



(c) Spent Fuel demand and supply plot.

Figure 3: Transition Scenario: Sinusoidal Power Demand.

4 Transition Scenarios

The objective of this section was to carry out various simulations to validate `D3ploy`'s current capabilities for simulating complex cycles. The Idaho National Laboratory Nuclear Fuel Cycle Evaluation and Screening Report [3] established several fuel cycle scenarios. As part of the project NEUP-FY16-10512, the simulations focused on the cases EG01, EG23, EG24. The scenarios started at EG01 – representing the current U.S. fuel cycle – and transitioned to advanced fuel cycles. The simulations utilized `d3ploy`'s NO, DO, and SO algorithms.

All the analyzed scenarios started at EG01. In EG01 all reactors were LWRs running a once-through cycle burning enriched-U. In EG23 fast reactors (FRs) produced all the power, relying on the continuous recycle of U/Pu supplemented by the addition of new natural-U to the cycle. EG24 was similar to EG23, but its cycle utilized continuous recycling of U/TRU with the addition of new natural-U.

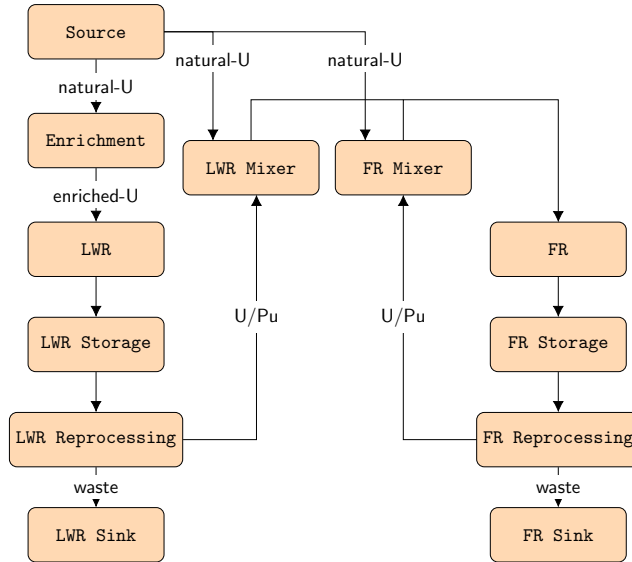
The present work focused on two transition scenarios: EG01-EG23 and EG01-EG24, as shown in Figure 4. The simulations started with a fleet of LWRs. After 80 years, the simulation progressively decommissioned the LWRs while transitioning to FRs. By the end of the cycle, all power was produced by FRs. Initial fueling of the FRs relied on reprocessed Pu from the LWR fleet. Following the transition, the FRs were able to produce their own Pu to sustain the cycle.

The following section presents the results for EG01-EG23 and EG01-EG24. The power demand was set at a constant 60 GW at all times. The transition scenarios used the capability of deploying facilities based on the difference between predicted demand and predicted supply, using a power supply buffer of 2000 MW.

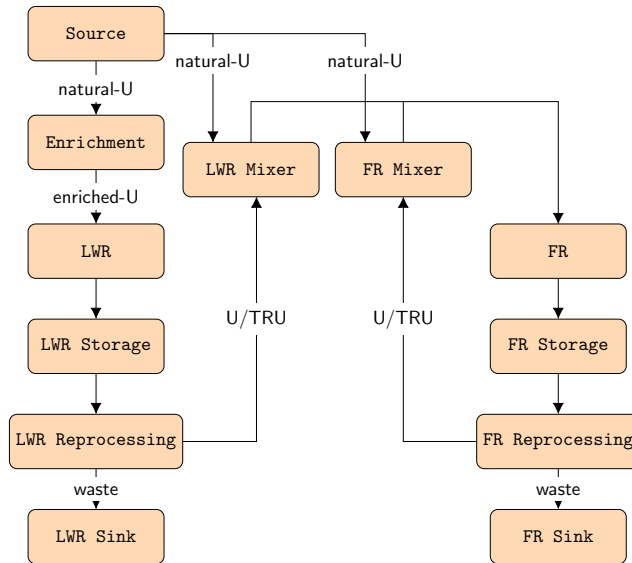
This section also includes a sensitivity analysis of the buffer size. A separate sensitivity analysis shows the dependency of the undersupply on the number of previous time steps used to calculate the predicted demand and supply.

4.1 EG01-EG23

Figure 5 shows the power demand and supply obtained using different prediction methods. Following it, Table 6 displays a comparison of the different algorithms. Table 6 displays the Cumulative Undersupply and the Cumulative Oversupply magnitudes. These values represent the summation of the difference between the power supplied and the power demanded for all the



(a) EG01-EG23.



(b) EG01-EG24.

Figure 4: Diagrams with facilities and mass flow of the scenarios EG01-EG23 and EG01-EG24.

time steps in the simulation. This magnitude could best be thought of as energy. For undersupply conditions, the magnitude represents lack of energy provided during the time steps where the supply did not meet the demand. Likewise, the oversupply would be the magnitude of excess energy produced.

Table 6: Undersupply and oversupply of Power for the different algorithms used to calculate EG01-EG23.

Algorithm	Power		
	No. of time steps of undersupply	Cumulative Undersupply[GW]	Cumulative Oversupply[GW]
MA	20	20.0	920.5
ARMA	18	7.7	1036.5
ARCH	0	0	1320.1
POLY	1	0.3	1783.5
EXP_SMOOTHING	20	11.0	1473.5
HOLT_WINTERS	20	11.0	1473.5
FFT	2	60.3	1751.9
SW_SEASONAL	20	18.6	1119.9

Table 7: No. of time steps with undersupply and under capacity of various commodities for the different algorithms used to calculate EG01-EG23.

Algorithm	Undersupply			Undercapacity	
	Sourceout	Enrichmentout	FR fuel	LWR PU	FR PU
MA	0	0	0	1	1
ARMA	0	0	0	1	1
ARCH	0	0	0	1	1
POLY	0	0	0	1	1
EXP_SMOOTHING	0	0	0	1	1
HOLT_WINTERS	0	0	0	1	1
FFT	0	1	0	1	1
SW_SEASONAL	0	0	0	1	1

Table 7 presents the no. of time steps with undersupply of natural-U (sourceout), enriched-U (enrichmentout), and FR fuel. The table also displays the no. of time steps where the capacity of the LWR Mixer to process LWR Pu and the capacity of the FR Mixer to process FR Pu are not enough (undercapacity). In this table it is possible to note that there is

one time step of delay between the supply of Pu and the deployment of the respective mixer facility.

One of the methods that performs the better is ARCH. For this scenario and said method, Figure 6 presents some of the different supply and demand time series plots for various commodities.

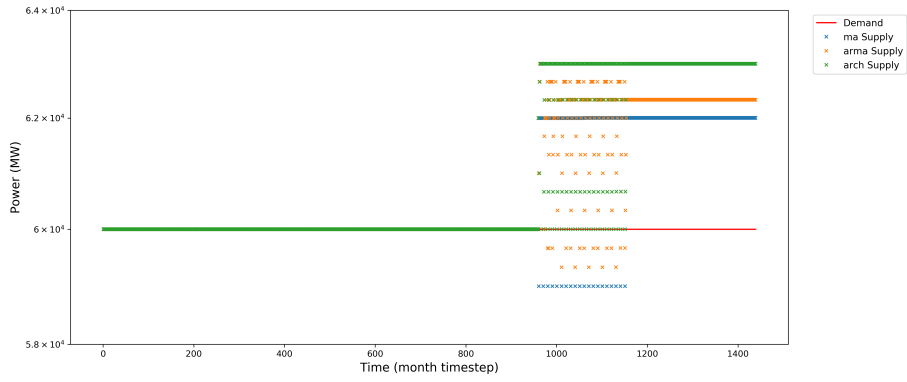
Figure 6a presents the number of Source facilities deployed, and the resultant demand and supply of natural-U. For this case, the capacity of natural-U supply is higher than the demand. It is easy to note that the demand in the beginning of the simulation is higher than in the end. The LWRs use enriched-U produced by the enrichment of natural-U, while the FRs require a smaller quantity of U for their fuel. Figure 6b displays the number of LWR Mixers deployed, and the supply and the capacity of LWR Pu (Pu produced by the LWRs). Logically, the supply of Pu decreases as the LWRs are decommissioned. Figure 6c shows the FR Mixers, and the supply and capacity of FR Pu. The supply of Pu increases as d3p1oy deploys new FRs.

4.2 EG01-EG24

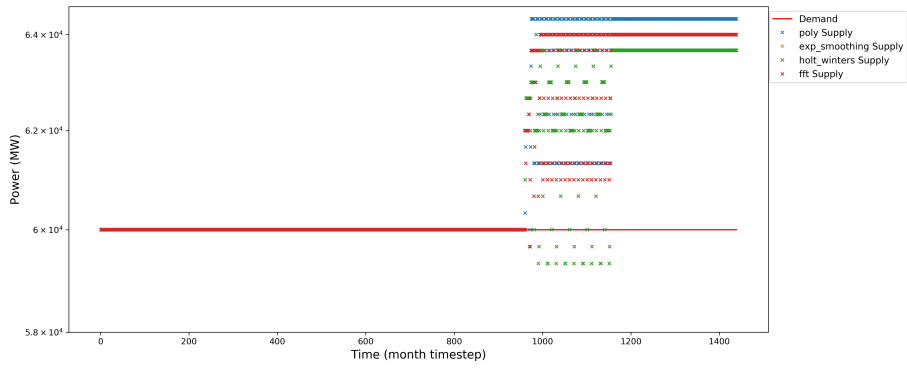
Figure 7 shows the power demand and supply obtained using different prediction methods. Following it, Tables 8 and 9 display a comparison of the different algorithms.

Table 8: Undersupply and oversupply of Power for the different algorithms used to calculate EG01-EG24.

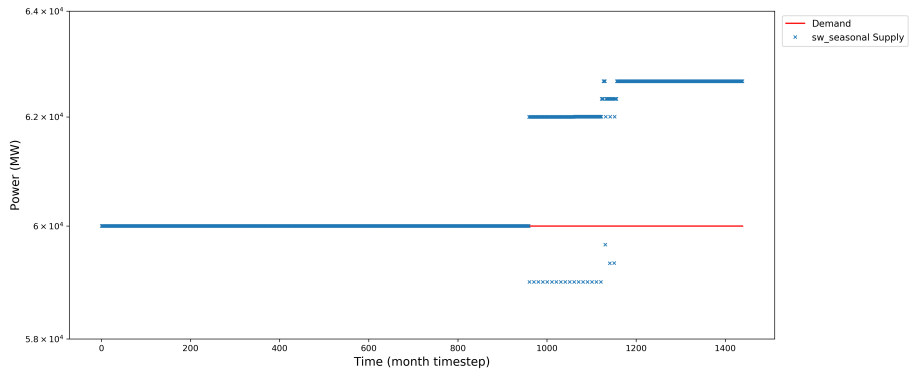
Algorithm	Power		
	No. of time steps of undersupply	Cumulative Undersupply[GW]	Cumulative Oversupply[GW]
MA	20	20.0	920.5
ARMA	18	7.7	1036.5
ARCH	0	0	1320.1
POLY	1	0.3	1783.5
EXP.SMOOTHING	20	11.0	1473.5
HOLT-WINTERS	20	11.0	1473.5
FFT	2	60.3	1751.9
SW_SEASONAL	20	18.6	1119.9



(a) NO algorithms.

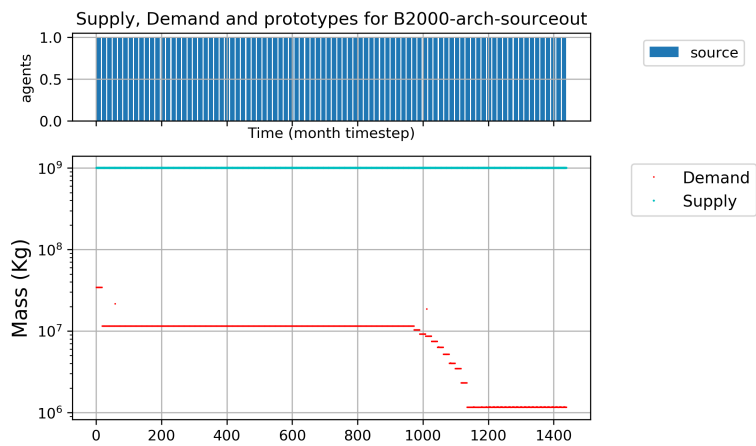


(b) DO algorithms.

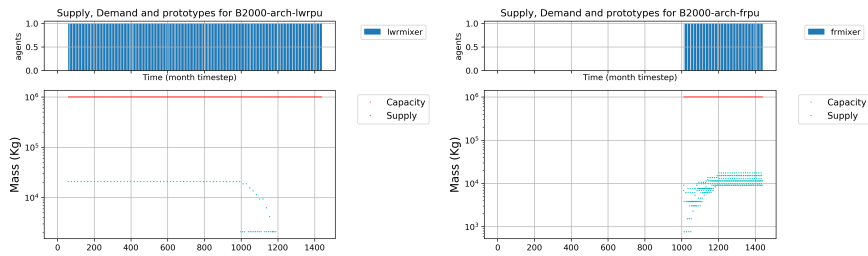


(c) SO algorithms.

Figure 5: Plot of the power demand and supply of EG01-EG23 for a constant power demand of 60GW for different prediction algorithms.

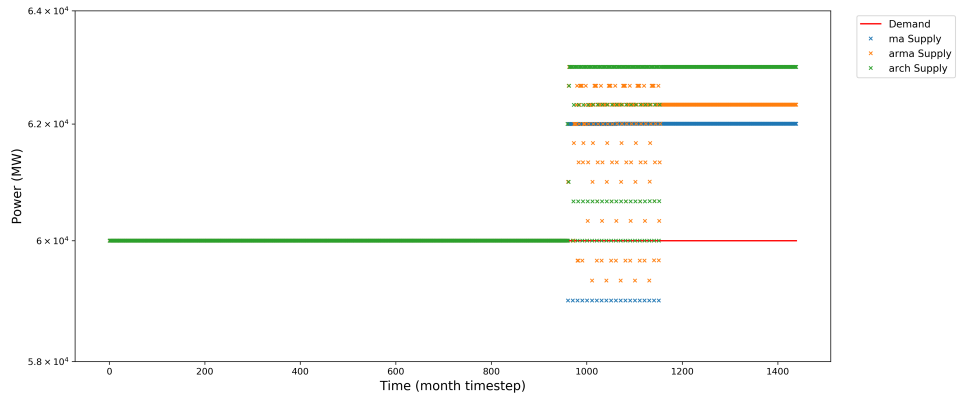


(a) Production of natural-U by the source.

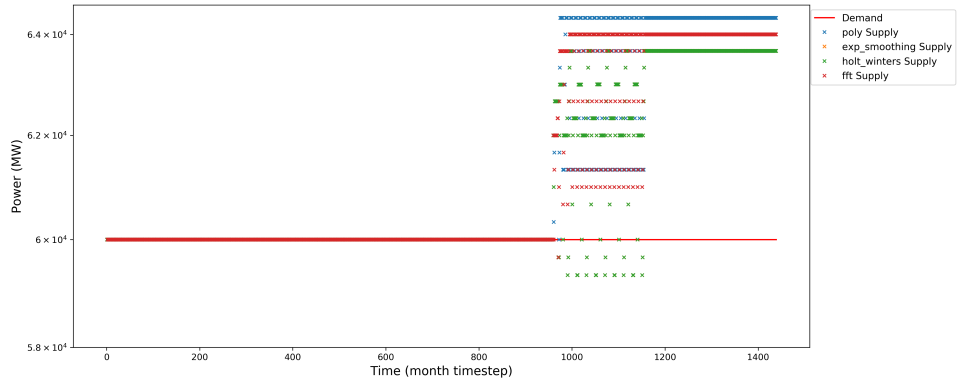


(b) Pu produced by the LWRs and (c) Pu produced by the FRs and exchanged to the LWR Mixer.

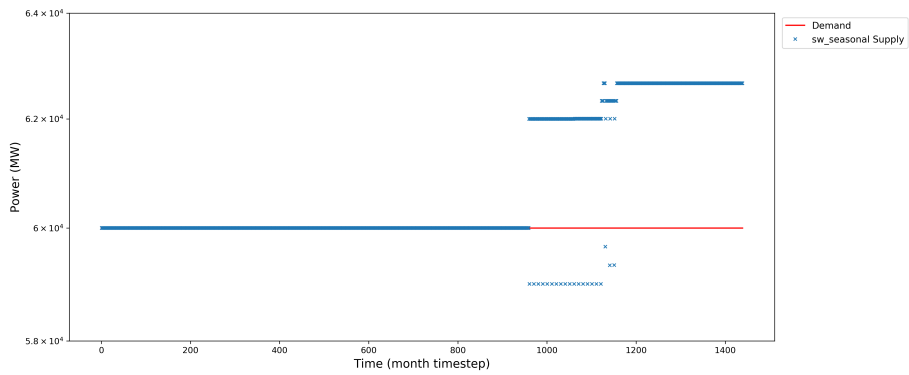
Figure 6: Plot for different commodities EG01-EG23.



(a) NO algorithms.



(b) DO algorithms.



(c) SO algorithms.

Figure 7: Plot of the power demand and supply of EG01-EG24 for a constant power demand of 60GW for different prediction algorithms.

Table 9: No. of time steps with undersupply and under capacity of various commodities for the different algorithms used to calculate EG01-EG24.

Algorithm	Undersupply			Undercapacity	
	Sourceout	Enrichmentout	FR fuel	LWR PU	FR PU
MA	0	0	0	1	1
ARMA	0	0	0	1	1
ARCH	0	0	0	1	1
POLY	0	0	0	1	1
EXP_SMOOTHING	0	0	0	1	1
HOLT_WINTERS	0	0	0	1	1
FFT	0	1	0	1	1
SW_SEASONAL	0	0	0	1	1

4.3 Buffer Size

This section focuses on the analysis of undersupply dependency on buffer size in the EG01-EG23 transition scenario. Table 10 shows the number of time steps that contain undersupply and the cumulative undersupply for different buffer sizes for various prediction methods. Figure 8 displays the cumulative undersupply as a function of buffer size.

Table 10: Dependency of the undersupply of Power on the buffer size.

Buffer [MW]	Algorithm	MA	ARMA	POLY	EXP_SMOOTHING	FFT
0	No. of time steps of undersupply	20	60	75	30	28
	Cumulative [GW]	60.0	87.3	52.9	68.3	93.3
2000	No. of time steps of undersupply	20	18	1	20	2
	Cumulative [GW]	20.0	7.7	0.3	11.0	60.3
4000	No. of time steps of undersupply	0	0	0	0	1
	Cumulative [GW]	0	0	0	0	60.

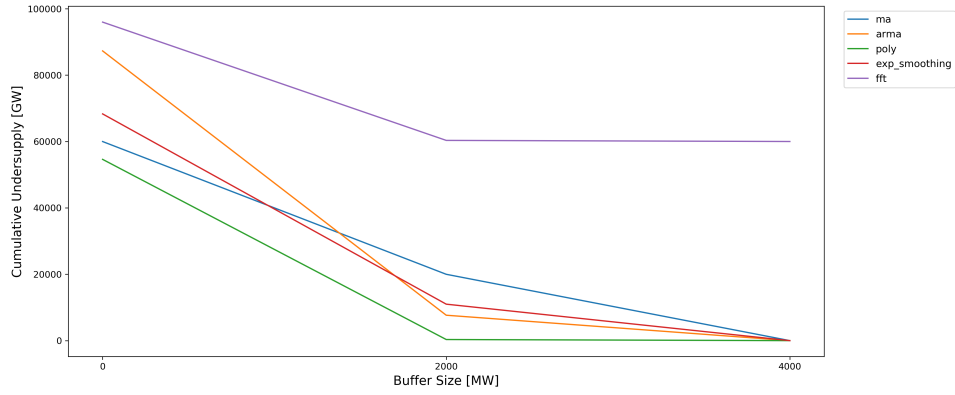


Figure 8: Plot of the dependency of the undersupply of Power on the buffer size.

4.4 Number of Forward Steps

This section focuses on the dependency on the number of forward steps calculated at each time step by the prediction methods in scenario EG01-EG23; the buffer size was fixed at 2000 MW. Table 11 shows number of time steps containing undersupply and the cumulative undersupply for different forward steps for some of the prediction methods. Figure 9 displays the cumulative undersupply as a function of the no. of forward steps.

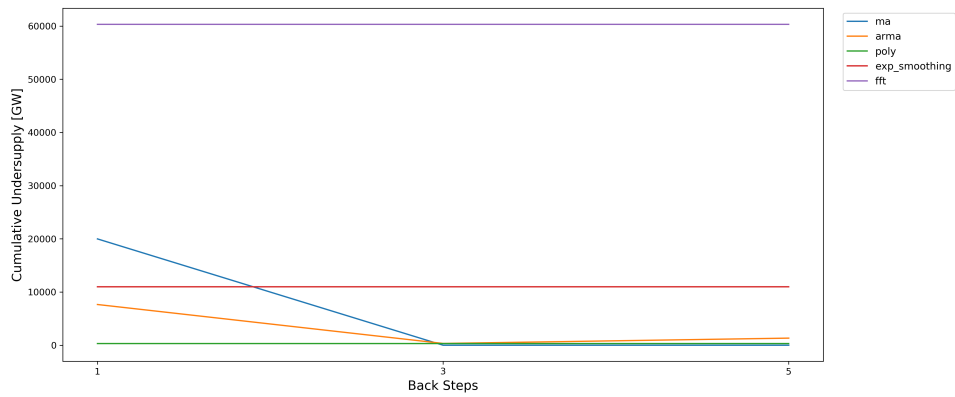


Figure 9: Plot of the dependency of the undersupply of Power on the no. of forward steps.

Table 11: Dependency of the undersupply of Power on the no. of forward steps.

No. of forward steps	Algorithm	MA	ARMA	POLY	EXP_SMOOTHING	FFT
1	No. of time steps of undersupply	18	20	2	20	1
	Cumulative [GW]	7.6	11.0	60.3	20.0	0.3
3	No. of time steps of undersupply	1	20	2	0	1
	Cumulative [GW]	0.3	11.0	60.3	0	0.3
5	No. of time steps of undersupply	4	20	20	0	1
	Cumulative [GW]	1.3	11.0	60.3	0	0.3

5 Conclusion and Next Steps

This paper describes the capabilities of `d3ploy` and demonstrates the use of `d3ploy` for simple transition scenarios with constant, linearly increasing, and sinusoidal power demand. The demonstration goes further with the more complex transition scenarios EG01-EG23 and EG01-EG24. This paper also provides insights on parameter inputs to ease the setup of larger transition scenarios that may include numerous facilities.

Future work includes setup of similar power demand transition scenarios for extended nuclear fuel cycles incorporating multiple reactor designs that consequently use different types of fuel. Such cases are currently under study. [3] established the transition scenarios EG01-EG29 and EG01-EG30. These scenarios are more complex than the cases presented in this report and the distribution of fuel between different reactor technologies play a main role in the transition. Additionally, as seen during the demonstration of `d3ploy` capabilities, a Decommissioning capability is highly useful for the setup of several NFCs and is currently under development.

6 Acknowledgements

This research is funded by the Department of Energy (DOE) Office of Nuclear Energy’s Nuclear Energy University Program (Project 16-10512) ”Demand-Driven Cyncamore Archetypes”. The authors would like to thank

members of the Advanced Reactors and Fuel Cycles (ARFC) group at the University of Illinois at Urbana-Champaign. We also thank our colleagues from the CYCLUS community, particularly those in the University of Wisconsin Computational Nuclear Engineering Research Group (CNERG) and the University of South Carolina Energy Research Group (ERGS) for collaborative CYCLUS development.

References

- [1] Kathryn D Huff, Jin Whan Bae, Robert R Flanagan, and Anthony M Scopatz. Current Status of Predictive Transition Capability in Fuel Cycle Simulation. page 11, 2017.
- [2] Kathryn D. Huff, Matthew J. Gidden, Robert W. Carlsen, Robert R. Flanagan, Meghan B. McGarry, Arrielle C. Opotowsky, Erich A. Schneider, Anthony M. Scopatz, and Paul P. H. Wilson. Fundamental concepts in the Cyclus nuclear fuel cycle simulation framework. *Advances in Engineering Software*, 94:46–59, April 2016. arXiv: 1509.03604.
- [3] R Wigeland, T Taiwo, H Ludewig, M Todosow, W Halsey, J Gehin, R Jubin, J Buelte, S Stockinger, and K Jenni. Nuclear Fuel Cycle Evaluation and Screening – Final Report. *Final Report*, page 10, 2014.