NEUP

Nuclear Energy
University Programs

U.S. Department of Energy

Market-Based and System-Wide Fuel Cycle Optimization

Paul Wilson, Anthony Scopatz,

Matthew Gidden, Robert Carlsen

University of Wisconsin-Madison

Cyclus Optimization

There are two distinct parts of a fuel cycle calculation that may be optimized:

- Per-time step supply chain optimization performed by a Dynamic Resource Exchange (DRE) algorithm.
- Systemtic optimization of fuel cycle scenarios with respect to top-level metrics over the many realizations.

Dynamic Resource Exchange

Dynamic Resource Exchange

- DRE: Core algorithm for fuel cycle simulation
- Recomputed at each time step
- Solves economic problem dynamically; no hard-coded supply-demand behavior
- Enables complicated fuel cycles

DRE Phases

Request for Bids

Queries each requesting Agent in the simulation that ***demands*** a resource

Phase 1: Request for bids

DRE Phases

Request for Bids

Queries each requesting Agent in the simulation that ***demands*** a resource

Response to Request for Bids

Queries each responding Agent in the simulation that ***supplies*** a resource

Phase 1: Request for bids

Phase 2: Response to request for bids

DRE Phases

Preference Adjustment

Agent **reviews** all matches;
opportunity for
preference
adjustment

Solution

Matches **selected**
for satisfiable
requests

DRE Phases

Request for Bids

Queries each requesting Agent in the simulation that ***demands*** a resource

Response to Request for Bids

Queries each responding Agent in the simulation that ***supplies*** a resource

Preference Adjustment

Agent ***reviews*** all matches; opportunity for preference adjustment

Solution

Matches ***selected*** for satisfiable requests

Phase 1: Request for bids

Phase 2: Response to request for bids

Phase 3: Preference Adjustment

Phase 4:
Market
Resolution

DRE Phases

Request for Bids

Queries each requesting Agent in the simulation that ***demands*** a resource

Response to Request for Bids

Queries each responding Agent in the simulation that ***supplies*** a resource

Preference Adjustment

Agent ***reviews*** all matches; opportunity for preference adjustment

Solution

Matches ***selected*** for satisfiable requests

Trade Execution

Material ***transaction*** takes place

Phase 1: Request for bids

Phase 2: Response to request for bids

Phase 3: Preference Adjustment

Phase 4:
Market
Resolution

Market-Based Research Tasks

1. Market resolution & performance

- Different modeling choices $\Rightarrow$ different preferences & constraints
- Requires different solvers with different performance
- Trade-off between modeling accuracy & performance

2. Bid optimization

- Supplier may not be able to match request exactly
- Can bid some approximate match
- Must know requesters preferences to offer best match

Linear Programming (LP) Background

Minimization or
Maximization
Objectives

$$\left\{ \begin{array}{l} \min_x z = c^T x \end{array} \right.$$

Equality
Constraints

$$\left\{ \begin{array}{l} \text{s.t. } Ax \geq b \\ x \geq 0 \end{array} \right.$$

Variable	Description
c	Cost vector
x	Decision variable
A	Constraint matrix
b	Threshold vector

LP Formulation

Variable	Description
H, h	Commodities
I, i	Bids
J, j	Requests
K, k	Capacities
c	Cost of commodity
x	Decision variable
β	Capacity coefficient
s	Supply capacity
d	Demand capacity

$$\min_x z = \sum_{i \in I} \sum_{j \in J} \sum_{h \in H} c_{i,j}^h x_{i,j}^h$$

$$\text{s.t.} \quad \sum_{j \in J} \sum_{h \in H} \beta_{i,k} x_{i,j}^h \leq s_{i,k} \quad \forall k \in K_I, \forall i \in I$$

$$\sum_{i \in I} \sum_{h \in H} \beta_{j,k} x_{i,j}^h \geq d_{j,k} \quad \forall k \in K_J, \forall j \in J$$

$$x_{i,j}^h \geq 0 \quad \forall x \in X$$

LP Supply Constraint: General

Variable	Description
H, h	Commodities
I, i	Bids
J, j	Requests
K, k	Capacities
$\beta_{i,k}(q_i^h)$	Conversion function
x	Decision variable
s	Supply capacity

In general, constraints can be a function of the isotopic profile of the request (quality, q_j). The conversion function of a supplier would be a function of the quality ($\beta_{i,k}(q_i^h)$):

$$\sum_{j \in J} \beta_{i,k}(q_j^h) x_{i,j}^h \leq s_{i,k} \quad \forall k \in K_i^h, \forall i \in I, \forall h \in H$$

LP Supply Constraint: Example

Variable	Description
x	Decision variable
ε	Requested enrichment level
s	Supply capacity
f	Conversion function

$$\sum_{j \in J} f_{SWU}(\varepsilon_j) x_{i,j}^{EU} \leq s_{i,SWU}$$

$$\sum_{j \in J} f_{NU}(\varepsilon_j) x_{i,j}^{EU} \leq s_{i,NU}$$

Mixed Integer-Linear Programming (MILP)

Required to allow two groups of consumers:

1. those that require *exclusive* orders
2. those that allow *partial* orders

$$J = J_p \cup J_e$$

Introduce a **binary variable**, y_{ij}^h :

- 1 if consumer j is sent commodity h by supplier i
- restrict number of resource flows to consumer j to 1

$$\sum_{h \in H_j} \sum_{i \in I} y_{i,j}^h = 1 \quad \forall j \in J_e$$

Variable	Description
H, h	Commodities
I, i	Bids
J, j	Requests
y_{ij}^h	Binary variable

MILP Supply Constraint: General

$$\sum_{j \in J_p} \beta_{i,k}(q_j^h) x_{i,j}^h + \sum_{j \in J_e} \beta_{i,k}(q_j^h) y_{i,j}^h \tilde{x}_j^h \leq s_{i,k}^h \quad \forall i \in I, \forall k \in K_i^h, \forall h \in H$$

Variable	Description
H, h	Commodities
I, i	Bids
J, j	Requests
K, k	Capacities
$\beta_{i,k}(q_i^h)$	Conversion function
x, y	Decision variable
s	Supply capacity

MILP Formulation

Variable	Description
H, h	Commodities
I, i	Bids
J, j	Requests
K, k	Capacities
c	Cost of commodity
x, y	Decision variable
β	Capacity coefficient
s	Supply capacity
d	Demand capacity

$$\min_{x,y} z = \sum_{h \in H} \sum_{i \in I} \sum_{j \in J_p} c_{i,j}^h x_{i,j}^h + \sum_{h \in H} \sum_{i \in I} \sum_{j \in J_e} c_{i,j}^h y_{i,j}^h \tilde{x}_j^h$$

$$\text{s.t.} \quad \sum_{j \in J_p} \beta_{i,k}(q_j^h) x_{i,j}^h + \sum_{j \in J_e} \beta_{i,k}(q_j^h) y_{i,j}^h \tilde{x}_j^h \leq s_{i,k}^h$$

$$\forall i \in I, \forall k \in K_i^h, \forall h \in H$$

$$\sum_{i \in I} \sum_{h \in H_j} \beta_{i,k}(q_j^h) x_{i,j}^h \geq d_j(H_j)$$

$$\forall k \in K_j, \forall j \in J_p$$

$$\sum_{i \in I} \sum_{h \in H_j} \beta_{i,k}(q_j^h) y_{i,j}^h \tilde{x}_j^h \geq d_j(H_j)$$

$$\forall k \in K_j, \forall j \in J_e$$

$$\sum_{h \in H} \sum_{i \in I} y_{i,j}^h = 1$$

$$\forall j \in J_e$$

$$x_{i,j}^h \geq 0$$

$$\forall x \in X$$

$$y_{i,j}^h \in \{0, 1\}$$

$$\forall y \in Y$$

Greedy Solver Algorithm

```
order request portfolios by average preference;
forall the request portfolios do
    order requests by average preference;
    matched  $\leftarrow 0$ ;
    while  $matched \leq q_J$  and  $\exists$  a request do
        get next request;
        order arcs by preference;
        while  $matched \leq q_J$  and  $\exists$  an arc do
            get next arc;
            remaining  $\leftarrow q_J - matched$ ;
            to_match  $\leftarrow \min\{\text{remaining}, \text{Capacity}(\text{arc})\}$ ;
            matched  $\leftarrow matched + \text{to\_match}$ ;
        end
    end
end
```

Market Design & Performance

- Currently studying accuracy-performance trade-off
 - Few suppliers with 100s of requesters
 - e.g. Reactors requesting enriched fuel
 - Few requesters with 100s of suppliers
 - e.g. Reactors supplying spent fuel
- Impact of preference fidelity
 - Introduce coarse and fine location preferences
- Impact of exclusivity
 - Exclusive trades vs. partial trades

System-Wide Optimization

Optimization Needs

- Single Objective - expensive to evaluate
- Discrete Variables (100s)
- Black-box, derivative free
- Non-linear, discontinuous
- Linear constraints

- Algorithms
 - Swarm (particle swarm, ant colony, etc.)
 - Direct search (Nelder-Mead, pattern search, etc)
 - Evolutionary algorithms
 - Surrogate-based techniques (Kriging)

System-wide Research Tasks

1. Optimizer Comparison

- PSwarm
- DAKOTA JEGA

2. Transition optimization

- Define only end point system configuration
- Specify metrics and objective function
- Allow all deployment decisions to be found by optimizer

3. Disruption risk mitigation

- “Super”-objective function defined as variance in transition objective over many possible disruptions
- Optimizer minimizes variance

Comparing Solvers

PSWARM

DAKOTA/EA

- Pattern search with particle swarm
- Continuous variables
 - Modify objective function with rounding
 - Modify PSwarm to keep mesh integral
- Inequality constraints - stays inside feasible region
- Parallel on poll and swarm search

- Evolutionary algorithm
- Discrete variables
- Penalty-based linear constraints
- Parallel

Optimization Basics

Facility	LWR				Repository			Fuel Fab	Objective
Year	1	2	3	...	1	2	...	...	
Trial 1	5	1	3	...	0	1	...	...	233.6
Trial 2	3	1	2	...	0	0	...	...	

Optimizer

Distributed
computing for
>100 full
simulations per
minute!!

Transition Optimization

- 1-Pass MOX recycle
- DU & UOX $\Rightarrow$ infinite
- MOX fab $\Rightarrow$ COSI equivalence

Transition Optimization

Run	Waste Penalty	Repo Discount	% Batches using MOX	First Repo Built (month)
1	0.0	0.0	0	48
2	0.2	0.0	25.6	Never
3	0.2	1.0	0	24
4	0.4	0.05	25.7	96

Transition Optimization

Run 1

Run 2

Run 3

Run 4

Disruption Risk Mitigation

- Policy change
 - e.g. Fukushima
- Facility disruption
 - e.g. THORP shutdown
- Value change
 - e.g. economics vs. waste

Disruption Risk Mitigation

- Optimum path for given disruption time
 - may perform poorly at other disruption times
- Some path may perform adequately over all possible disruption times
 - Minimum performance?
 - Variance in performance?

Optimization Basics

Facility	LWR				Repository			Fuel Fab	Objective
Year	1	2	3	...	1	2	...	...	
Trial 1	5	1	3	...	0	1	...	...	233.6
Trial 2	3	1	2	...	0	0	...	...	

Optimizer

Questions?