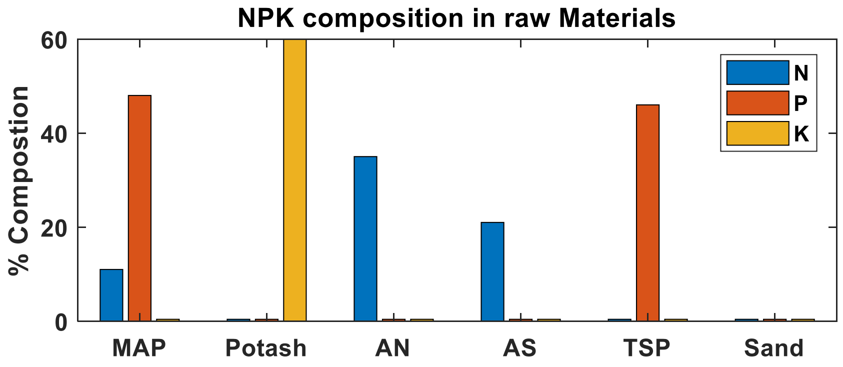
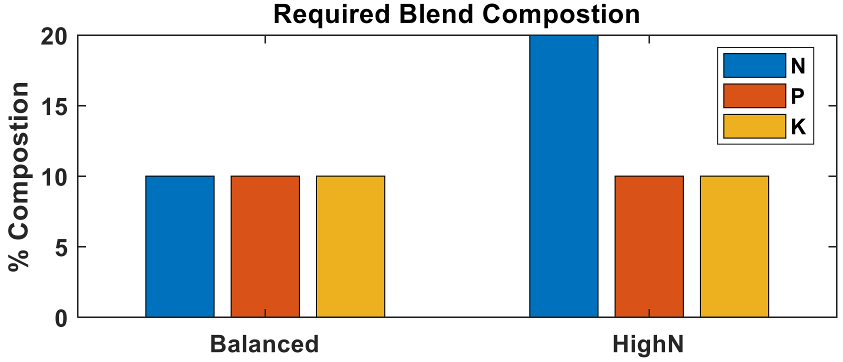
MATLAB EXPO 2018

Developing Optimization Algorithms for Real-World Applications

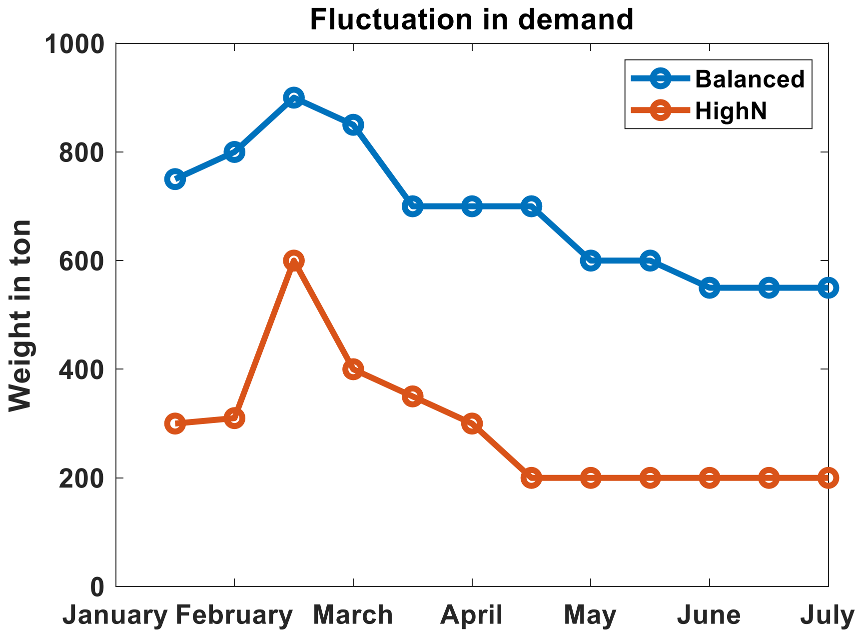
Gautam Ponnappa PC – Training Engineer
Viju Ravichandran, PhD – Education Technical Evangelist






```
disp(rawCost)
```

	MAP	Potash	AN	AS	TSP	Sand
January	350	610	300	135	250	80
February	360	630	300	140	275	80
March	350	630	300	135	275	80
April	350	610	300	125	250	80
May	320	600	300	125	250	80
June	320	600	300	125	250	80
July	320	600	300	125	250	80
August	320	600	300	125	240	80
September	320	600	300	125	240	80
October	310	600	300	125	240	80
November	310	600	300	125	240	80
December	340	600	300	125	240	80



“For a given ***system****, it is the selection of a best element, with regard to some criteria, to achieve best results”*

Optimization

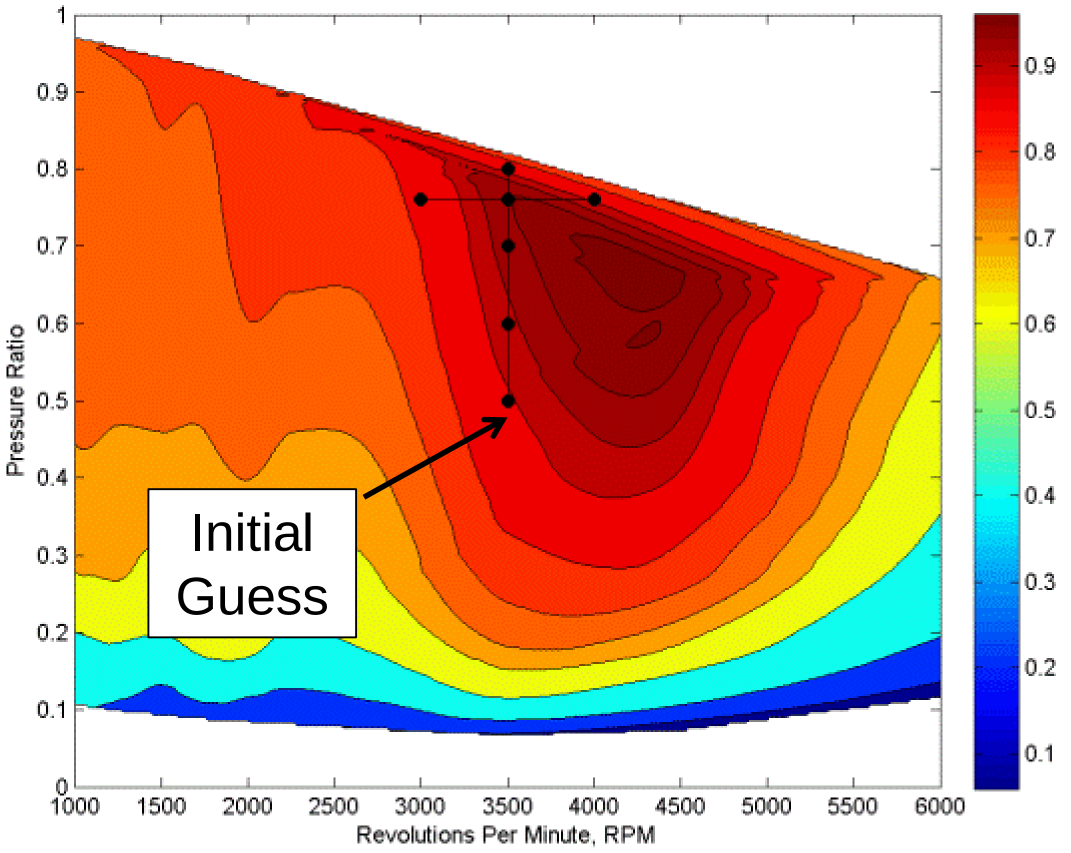
*“For a given **system**, it is the selection of a best element, with regard to some criteria, to achieve **optimal** results”*

Agenda

- Why Optimization?
- Optimization Workflow
- Problem Formulation using MATLAB
  - Multiperiod Production Planning
  - Need for Discrete Event Simulation
  - Batch Production Process

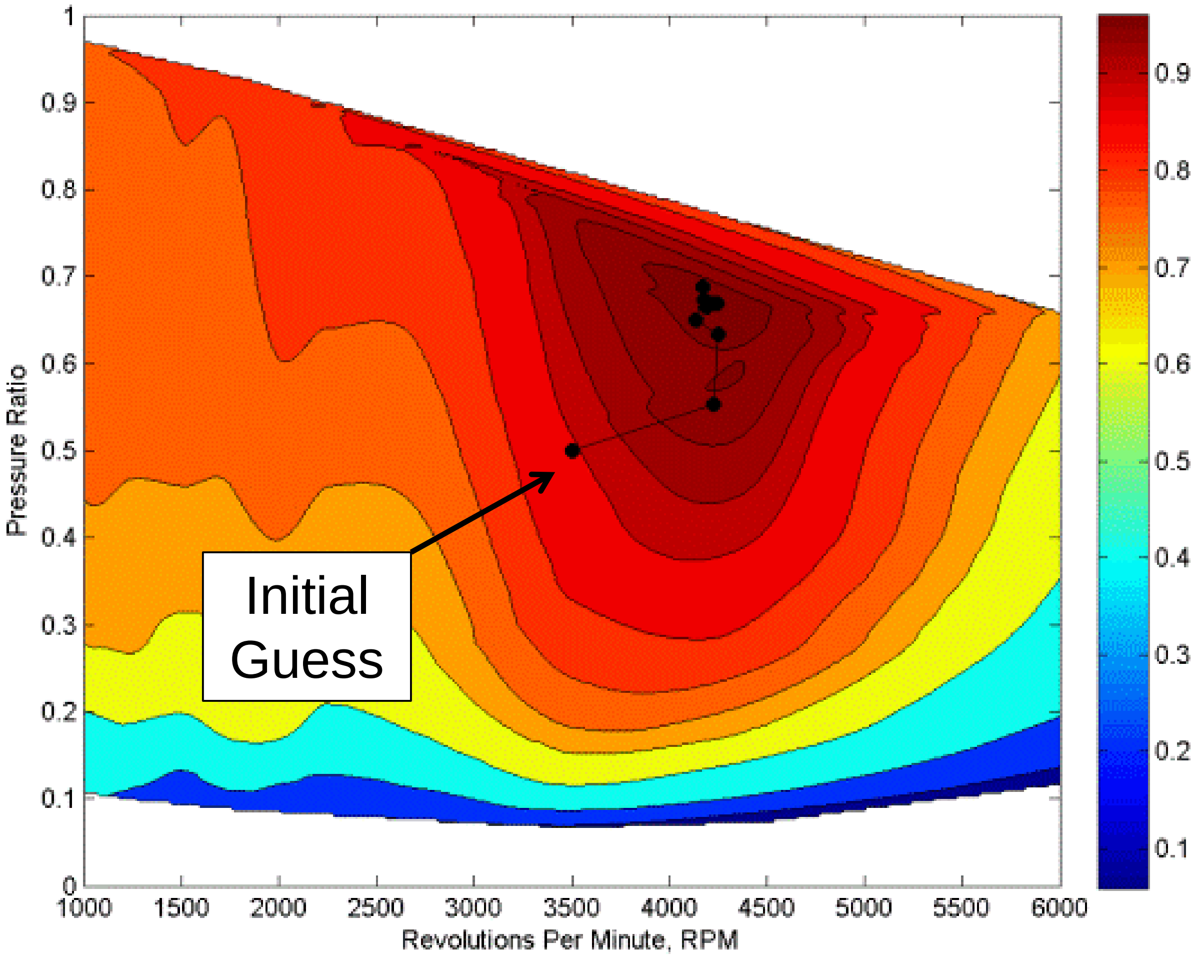
Why use optimization?

Manually (trial-and-error or iteratively)



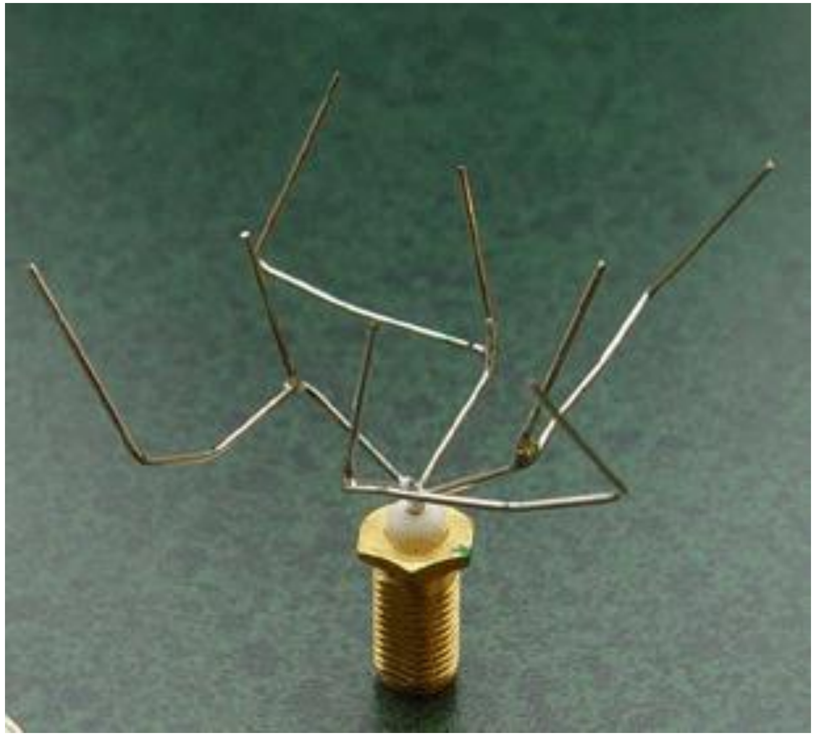
Why use optimization?

Automatically (using optimization techniques)



Why use optimization?

- Finding better (optimal) designs
- Faster Design Evaluations
- Useful for trade-off analysis
- Non-intuitive designs may be found

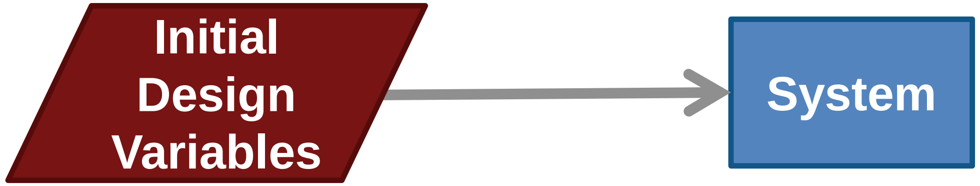


Antenna Design Using Genetic Algorithm
http://ic.arc.nasa.gov/projects/esg/research/antenna.htm

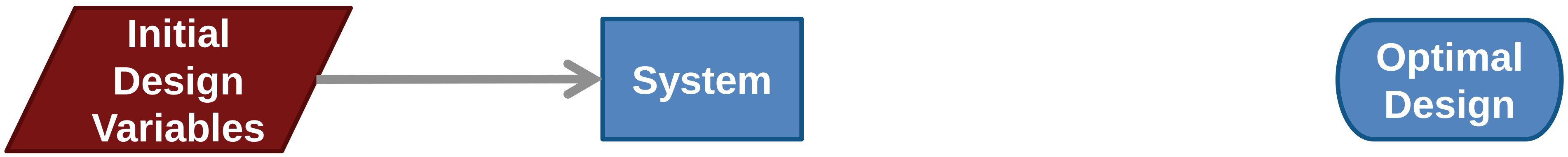
Optimization Workflow

System

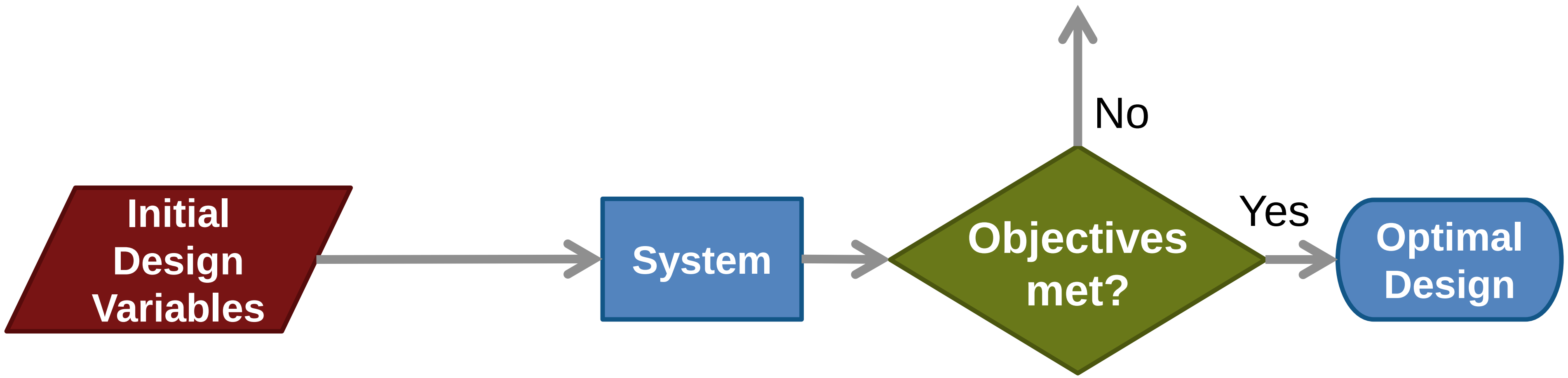
Optimization Workflow



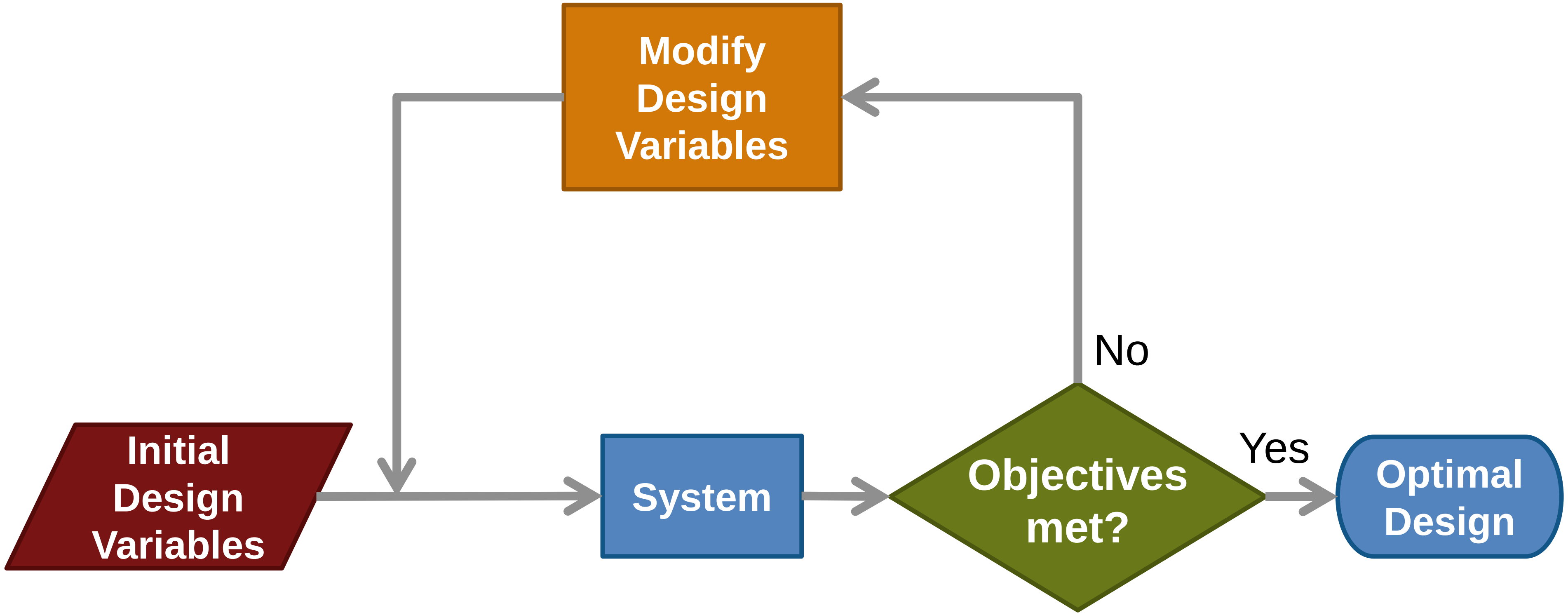
Optimization Workflow



Optimization Workflow



Optimization Workflow



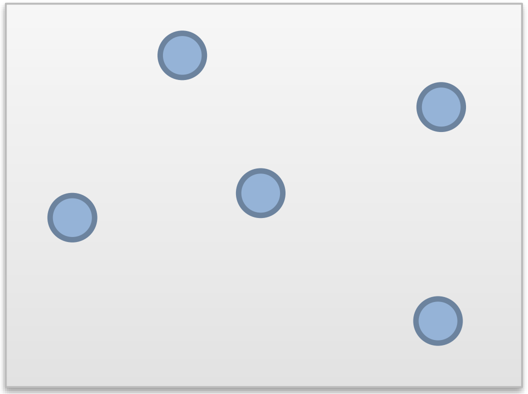
Traveling Salesman Problem

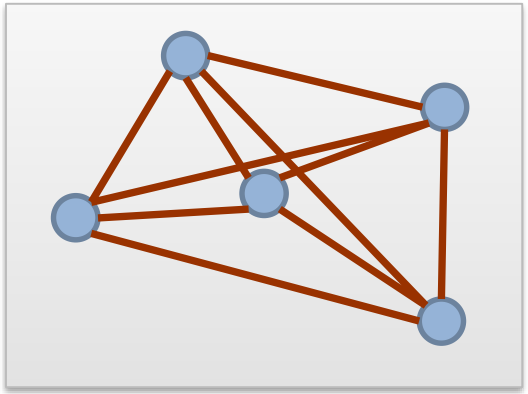
Problem

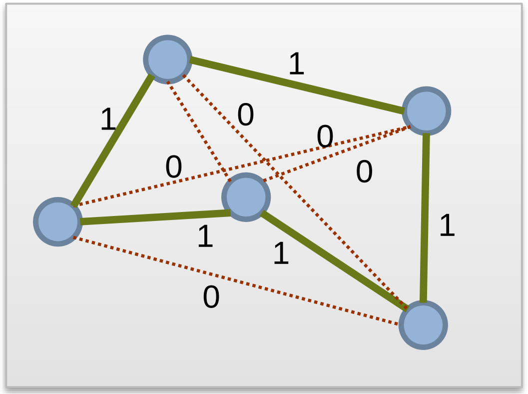
- How to find the shortest path through a series of points?

Solution

- Calculate distances between all combinations of points
- Solve an optimization problem where variables correspond to trips between two points







Optimization Solution for Traveling Salesman Problem

Decision variables: Binary vector based on whether the trip exists or not

Objective: Minimize the distance traveled

Constraints: Each stop on only two trips

Solver: Integer Linear Programming Algorithm

```
opts = optimoptions('intlinprog','Display','none');
[xopt,costopt,exitflag,output] = intlinprog(dist,intcon,[],[],Aeq,beq,lb,ub,opts);
```

Optimization Solution for Traveling Salesman Problem

Optimization Solution for Traveling Salesman Problem

Data Analytics Workflow

Access and Explore Data
- Files
- Databases
- Sensors

Preprocess Data
- Working with Messy Data
- Data Reduction/ Transformation
- Feature Extraction

Develop Predictive Models
- Model Creation e.g. Machine Learning
- Hyperparameter Optimization
- Model Validation

Develop Prescriptive Models
- Optimal Decisions
- Constraints
- What-if Analysis

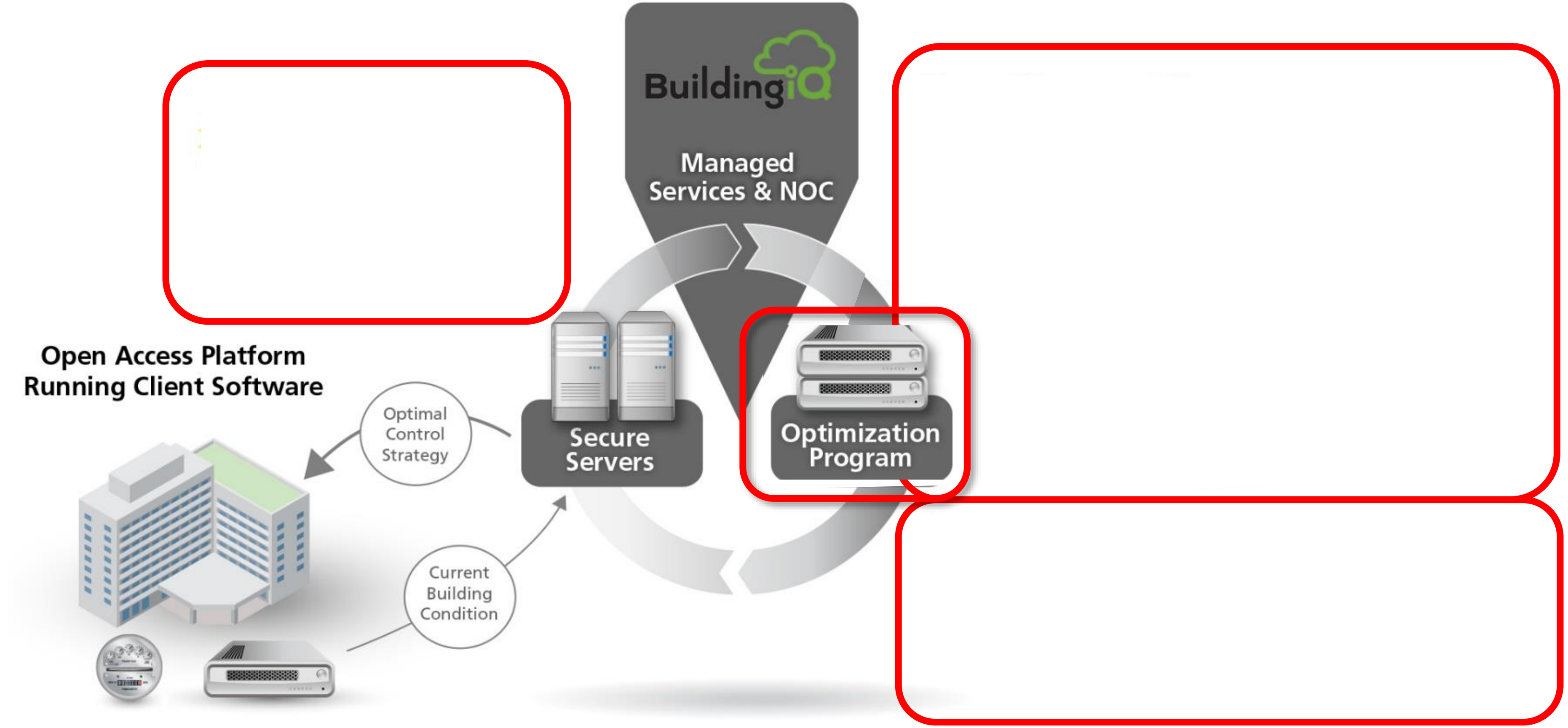
Integrate Analytics with Systems
- Desktop Apps
- Enterprise Scale Systems
  - MATLAB, Excel, .NET, .exe, C/C++, Java, .dll
- Embedded Devices and Hardware

Steps in Optimization Modeling

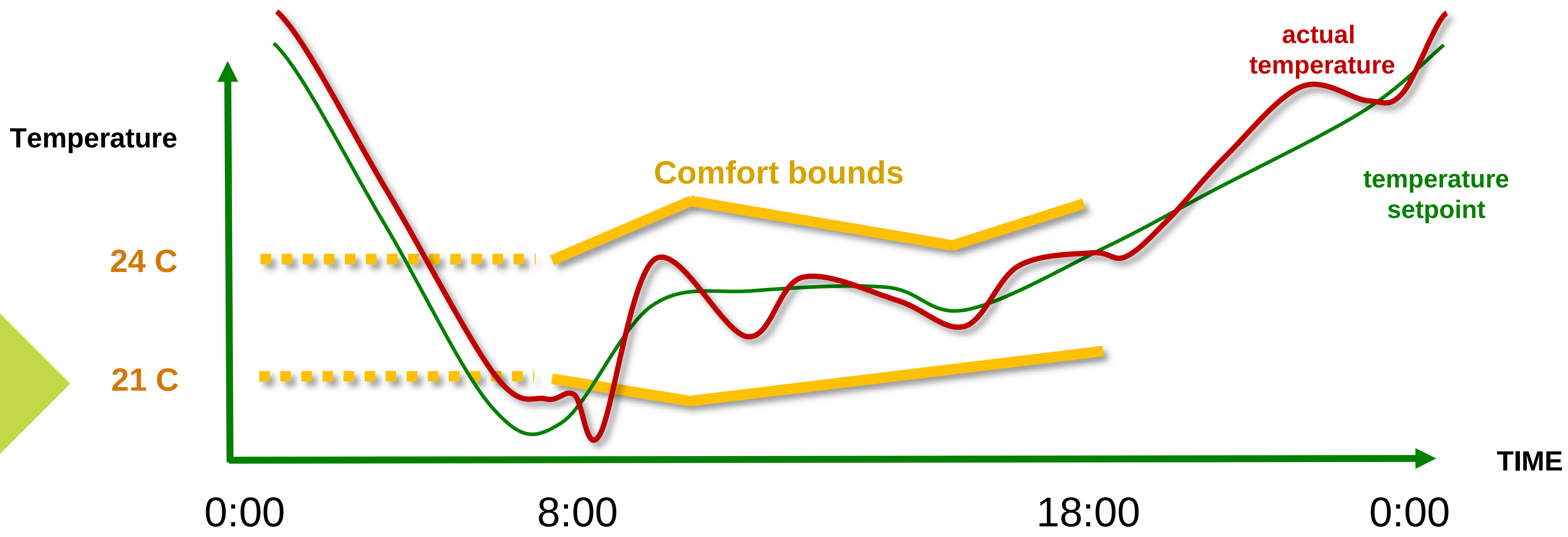
- Get overall idea of the system
- What is the goal? What are you trying to achieve?
- Identify variables
- Identify constraints
- Identify the inputs and outputs you can control
- Specify all quantities mathematically
- Check the model for completeness and correctness

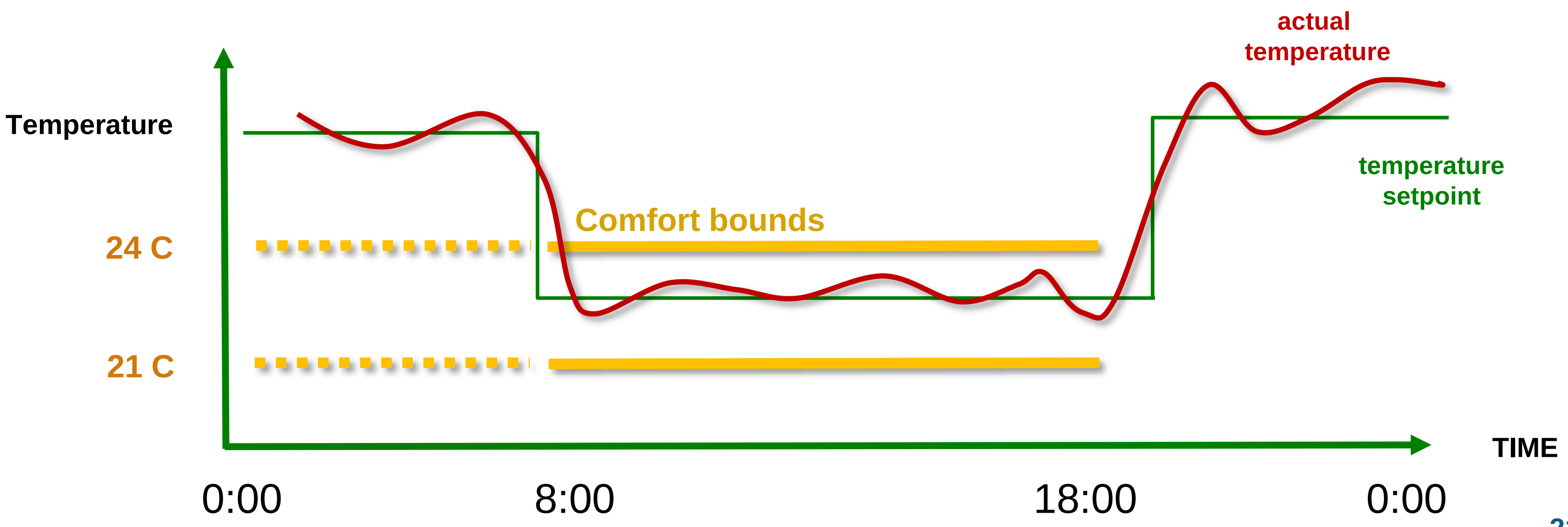
BuildingiQ

Adaptive building energy management

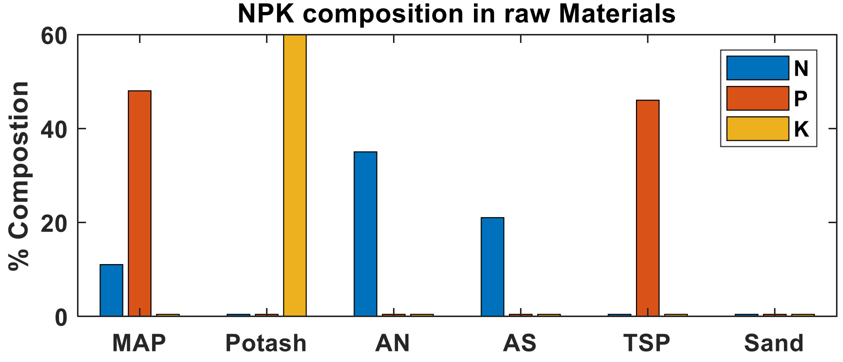


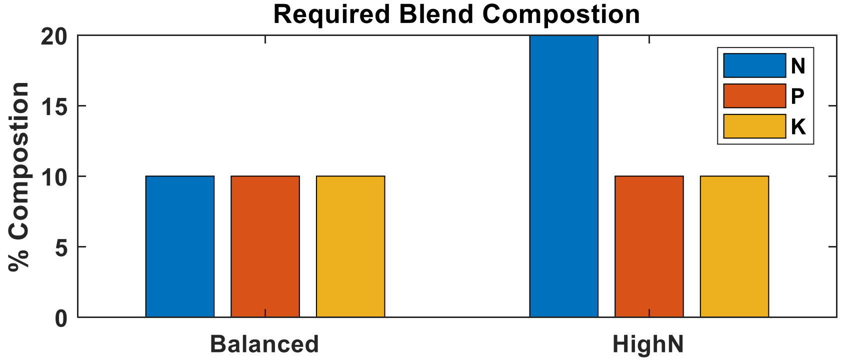
25% cost reduction





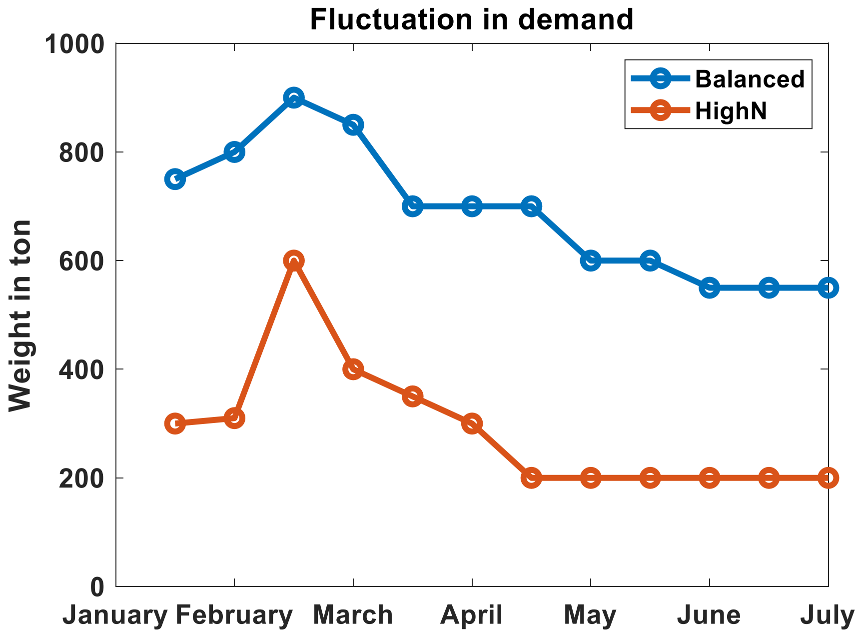
Production Plant Case Study



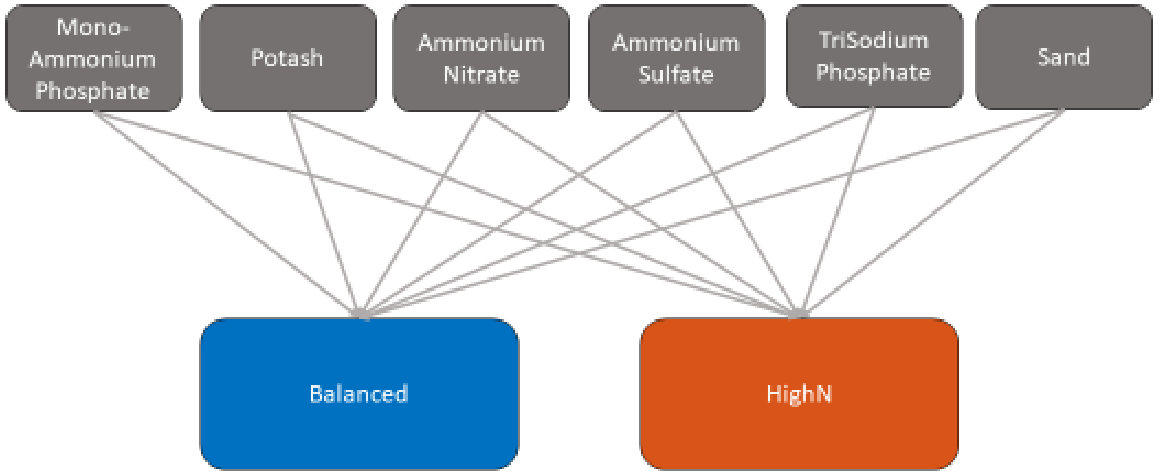



```
disp(rawCost)
```

	MAP	Potash	AN	AS	TSP	Sand
January	350	610	300	135	250	80
February	360	630	300	140	275	80
March	350	630	300	135	275	80
April	350	610	300	125	250	80
May	320	600	300	125	250	80
June	320	600	300	125	250	80
July	320	600	300	125	250	80
August	320	600	300	125	240	80
September	320	600	300	125	240	80
October	310	600	300	125	240	80
November	310	600	300	125	240	80
December	340	600	300	125	240	80



Objective: To maximize profits while meeting demand



	MAP	Potash	AN	AS	TSP	Sand
January	350	610	300	135	250	80
February	360	630	300	140	275	80
March	350	630	300	135	275	80
April	350	610	300	125	250	80
May	320	600	300	125	250	80
June	320	600	300	125	250	80
July	320	600	300	125	250	80
August	320	600	300	125	240	80
September	320	600	300	125	240	80
October	310	600	300	125	240	80
November	310	600	300	125	240	80
December	340	600	300	125	240	80

Balanced	HighN
400	550

Decision Variables

- Two decision variables
  - Quantities of fertilizer blends that you make and sell each month
  - Raw ingredients that we use to make those blends

```
make = optimvar('make',months,blends,'LowerBound',0);
sell = optimvar('sell',months,blends,'LowerBound',0,'UpperBound',blendDemand{months,blends});
use  = optimvar('use',months,raws,blends,'LowerBound',0);
```

- Additionally, create a variable that represents the inventory at each time.

```
inventory = optimvar('inventory',months,blends,'LowerBound',0,'UpperBound',inventoryCapacity);
```

Objective Function

- Objective function for this problem is profit, which we want to maximize

```
inventoryProblem = optimproblem('ObjectiveSense','maximize');
```

- To calculate the objective function in terms of the problem variables, calculate the revenue and costs.

```
inventoryProblem.Objective = revenue - ingredientCost - storageCost;
```

Constraints

- **Connection among production, sales, and inventory**
  - Final inventory is fixed
  - Total inventory at each time is bounded
  - Produce a limited amount in each time period

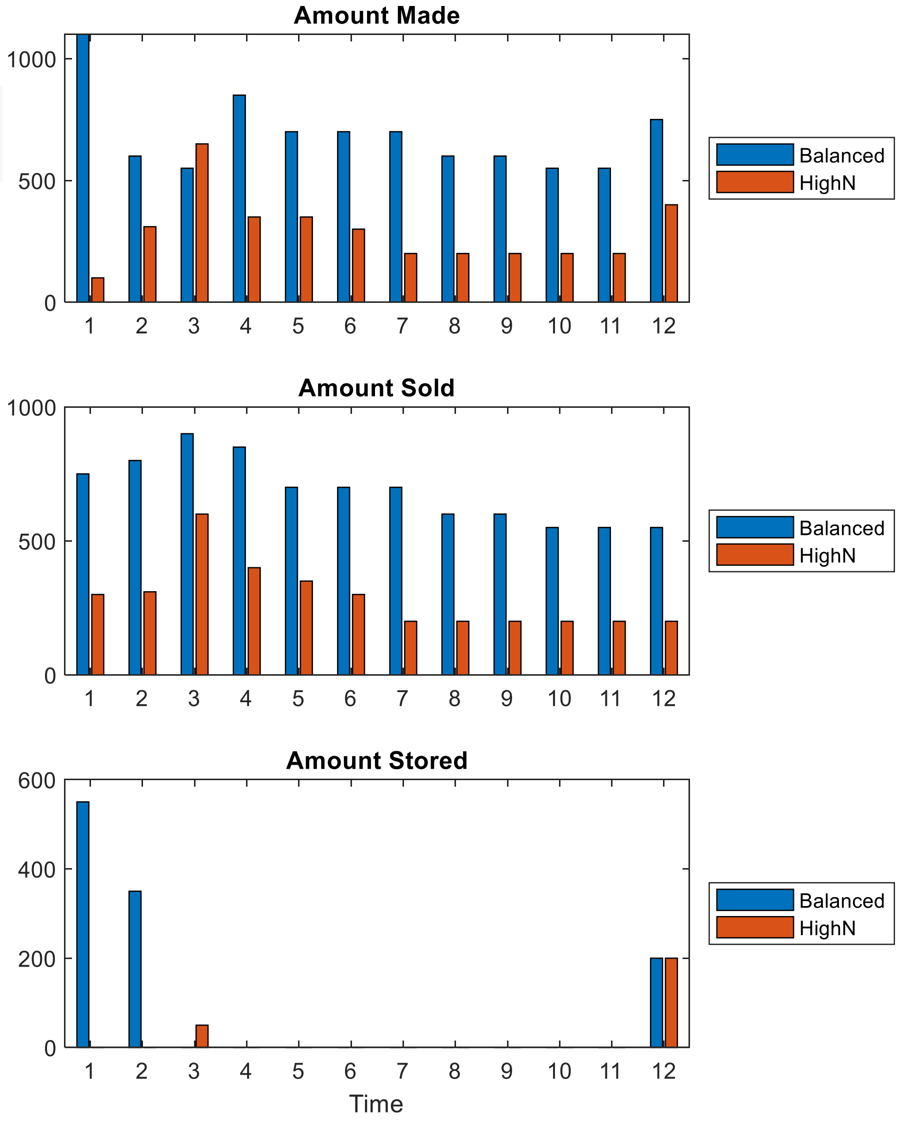
```
inventoryProblem.Constraints.materialBalance = materialBalance;
inventoryProblem.Constraints.finalInventory = finalInventory;
inventoryProblem.Constraints.inventoryLimit = inventoryLimit;
inventoryProblem.Constraints.processLimit = processLimit;
inventoryProblem.Constraints.rawMaterialUse = rawMaterialUse;
inventoryProblem.Constraints.blendNutrientsQuality = blendNutrientsQuality;
```

```
writeproblem(inventoryProblem,'inventoryProblem.txt')
```

Solution

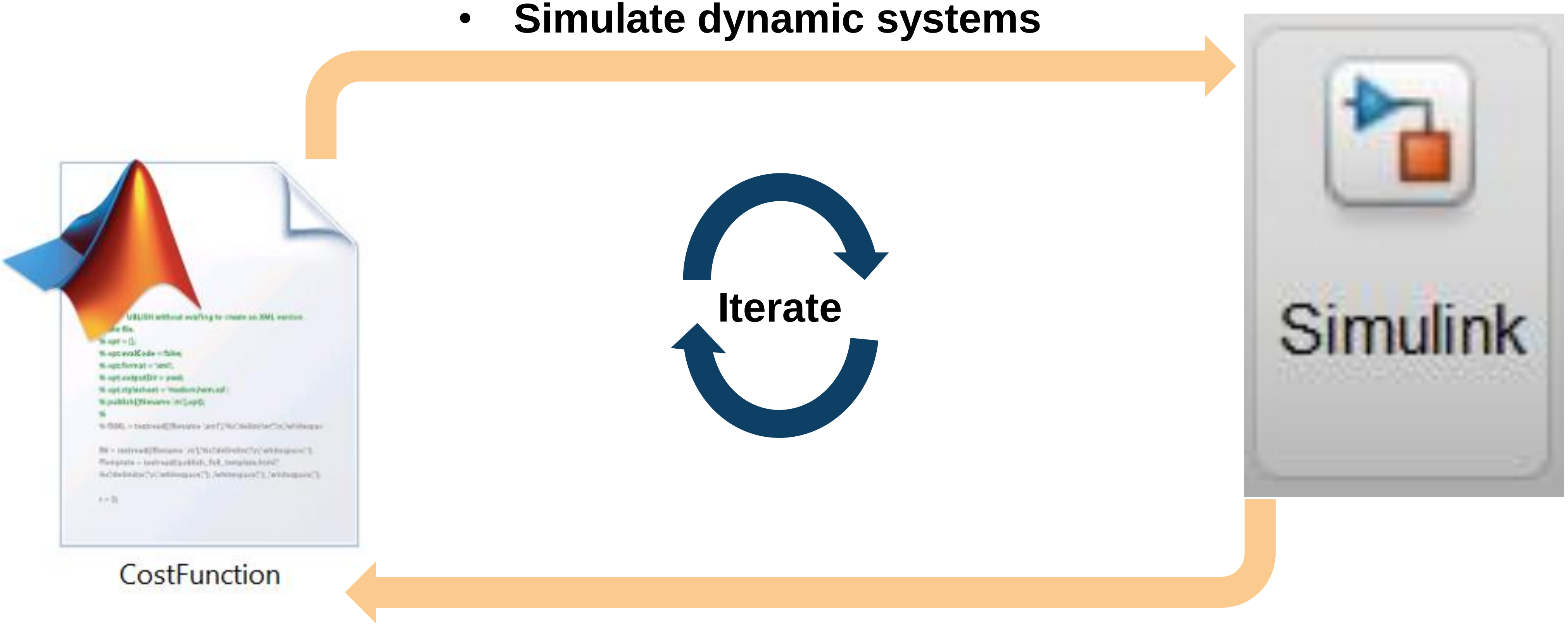
```
[sol,fval,exitflag,output] = solve(inventoryProblem)
```

	makeBalan...	makeHighN	sellBalanced	sellHighN	storeBalan...	storeHighN
1 January	1.1000e+03	100.0000	750	300	550.0000	0
2 February	600.0000	310.0000	800	310	350.0000	0
3 March	550.0000	650.0000	900	600	0	50.0000
4 April	850	350.0000	850	400	0	0
5 May	700	350.0000	700	350	0	0
6 June	700.0000	300.0000	700	300	0	0
7 July	700.0000	200.0000	700	200	0	0
8 August	600	200.0000	600	200	0	0
9 September	600	200.0000	600	200	0	0
10 October	550	200.0000	550	200	0	0
11 November	550	200.0000	550	200	0	0
12 December	750	400.0000	550	200	200.0000	200.0000



Any other approach?

- Initialise/Tune decision variables
- Simulate dynamic systems

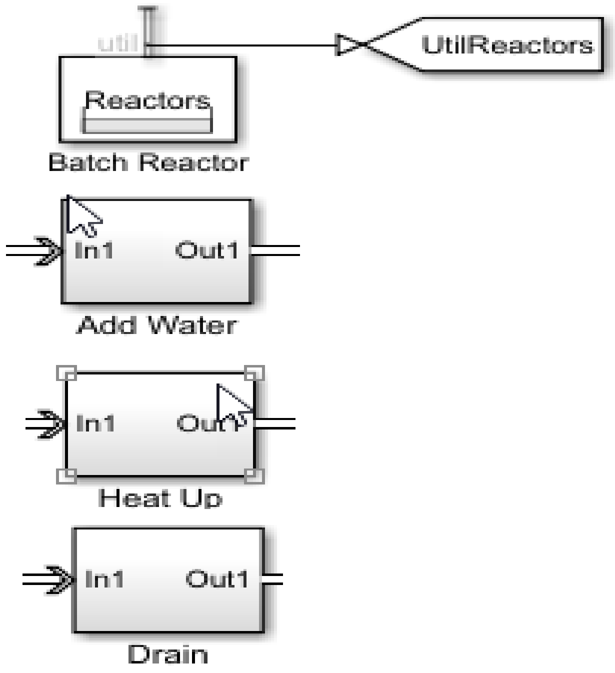


- Collect Simulation output
- Check if optimal criterion is met

Batch Production Process

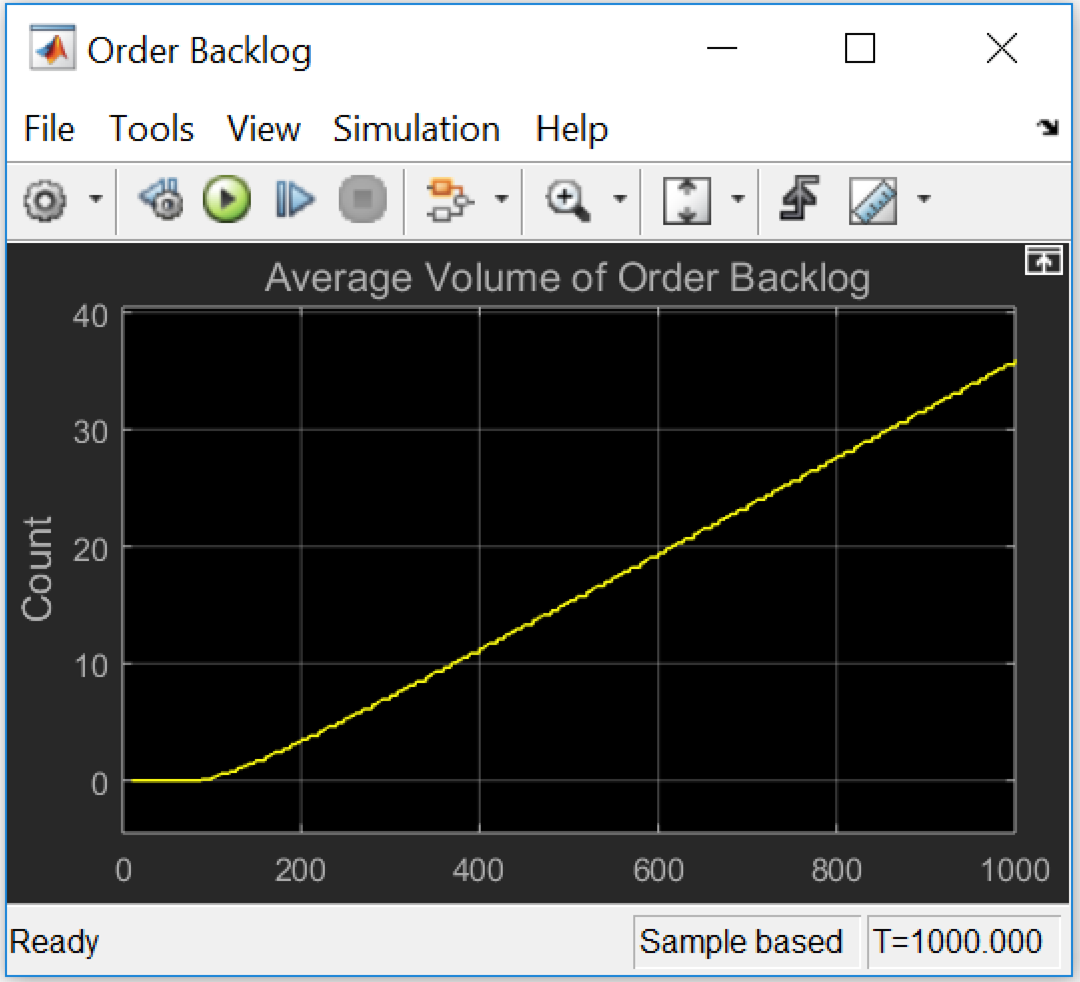
- Need to produce 2 types of chemical compounds in batches to meet incoming demand

- Production environment uses 4 shared resources(Reactors, water pumps, heaters and drains)



Batch Production Process

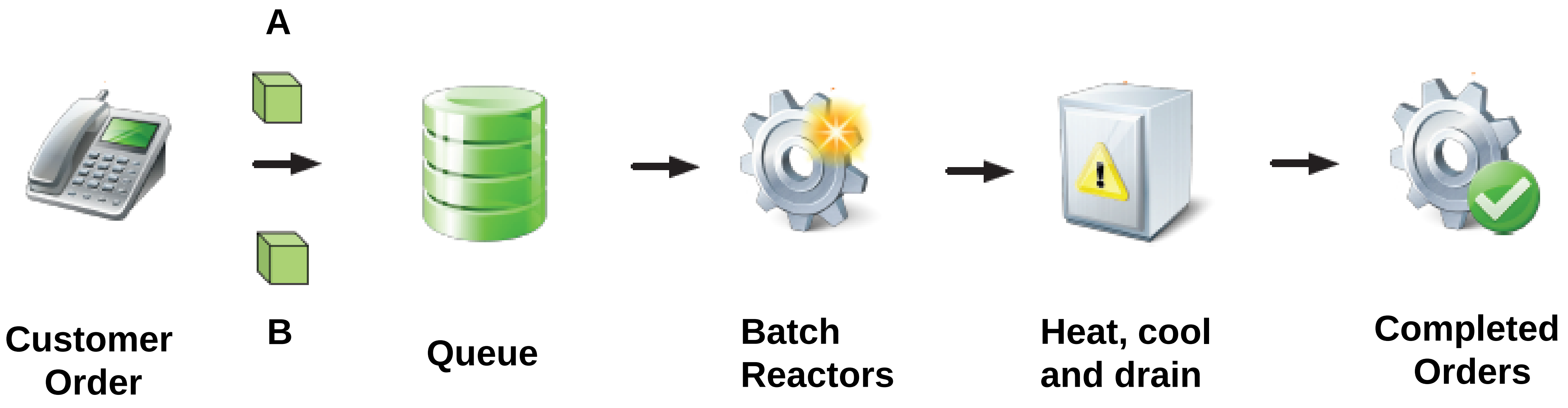
- Huge backlog of orders with current resources, not able to meet demand



Optimization Problem?

How many **batch reactors, water pumps, heaters and drains** do I need to purchase so that **backlog is minimized**, **without spending** a lot on my production environment?

Process cycle



Simulation environment

Time-Based Simulation: Simulink

- Integrators, Filters, Mathematical computation blocks
- Modeling environment based on a set of solvers for solving differential and/or difference equations

Event-Based Simulation: SimEvents

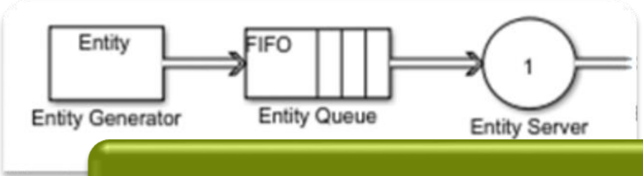
- Generators, Queues, Servers, and Router blocks
- Producing and processing entities (e.g. packets, planes, raw material)

Discrete Event Simulation

- Simulation of real-world processes wherein there might be a series of instantaneous occurrences(events), or discrete events to model behavior of processes
- Focus on service and transit time, utilization and throughput

Discrete-event modeling in Simulink

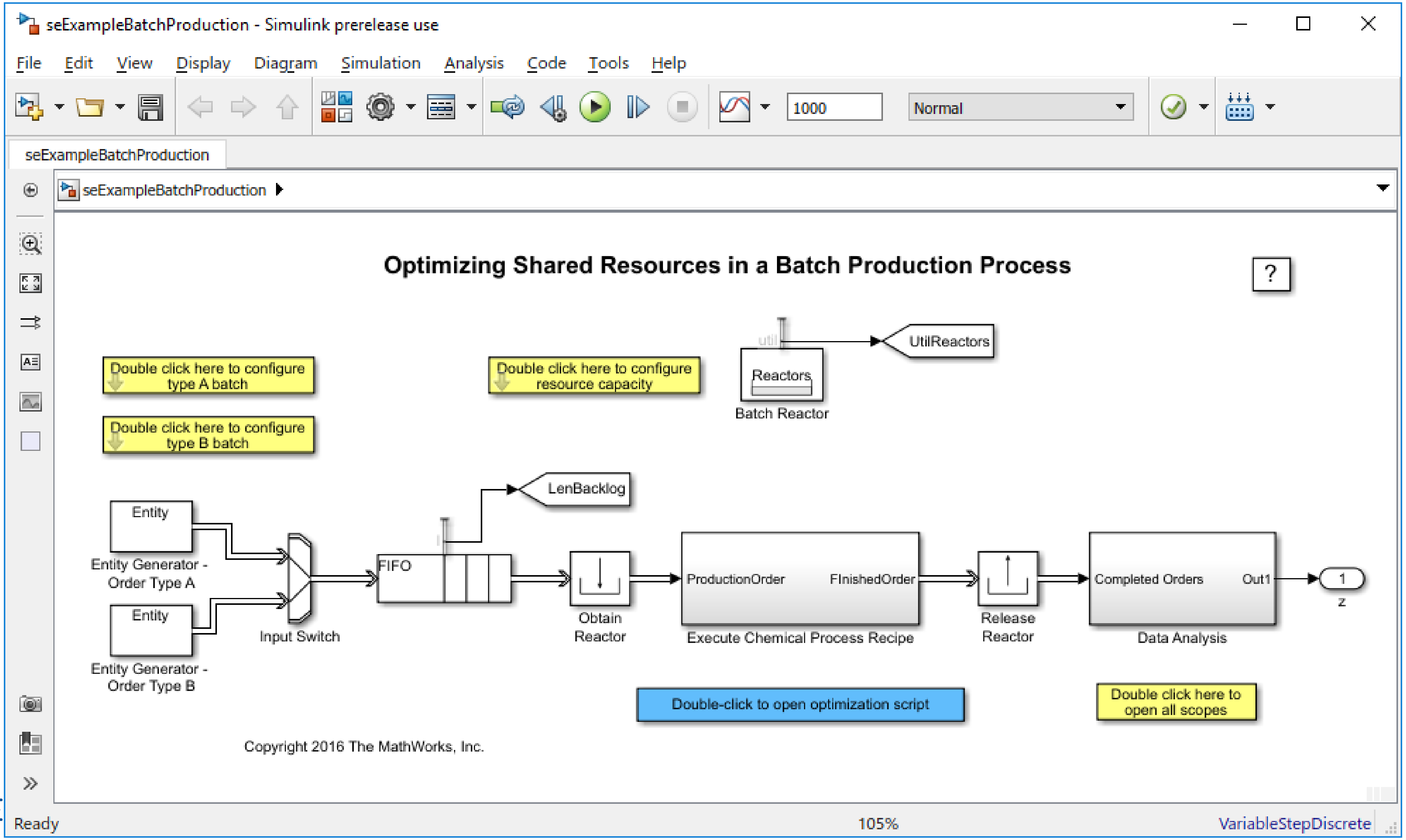
Production line



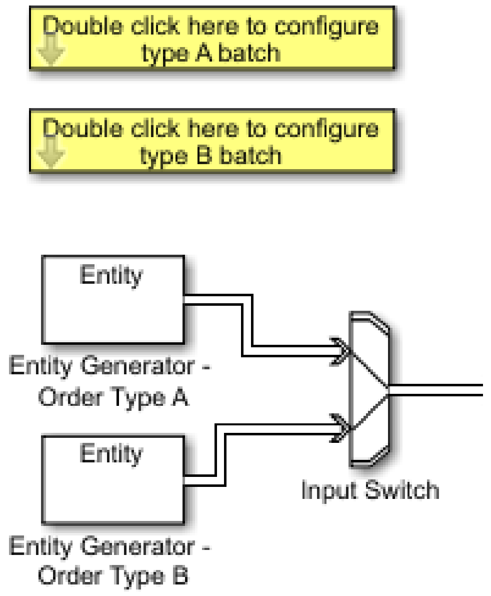
SimEvents

Production line	SimEvents
Order Sequence	Entities Attributes
Order queue Processing unit	Queue Server
Wait time Orders completed Orders not on time	Average wait statistic Entities departed Entities timed out
Maximize fulfillment Minimize wait time	Maximize throughput Minimize latency

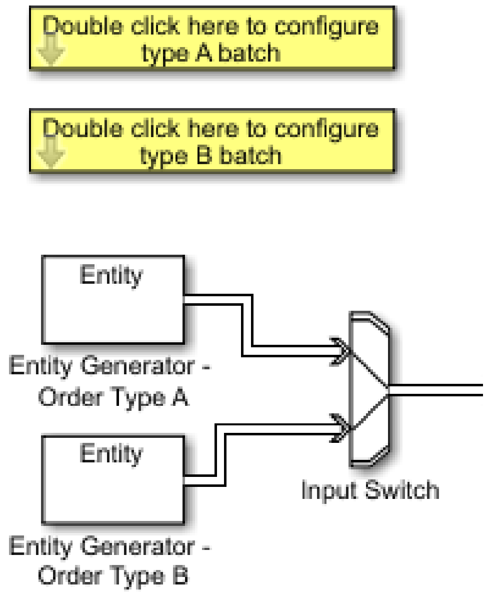
SimEvents Model for Batch Production Process

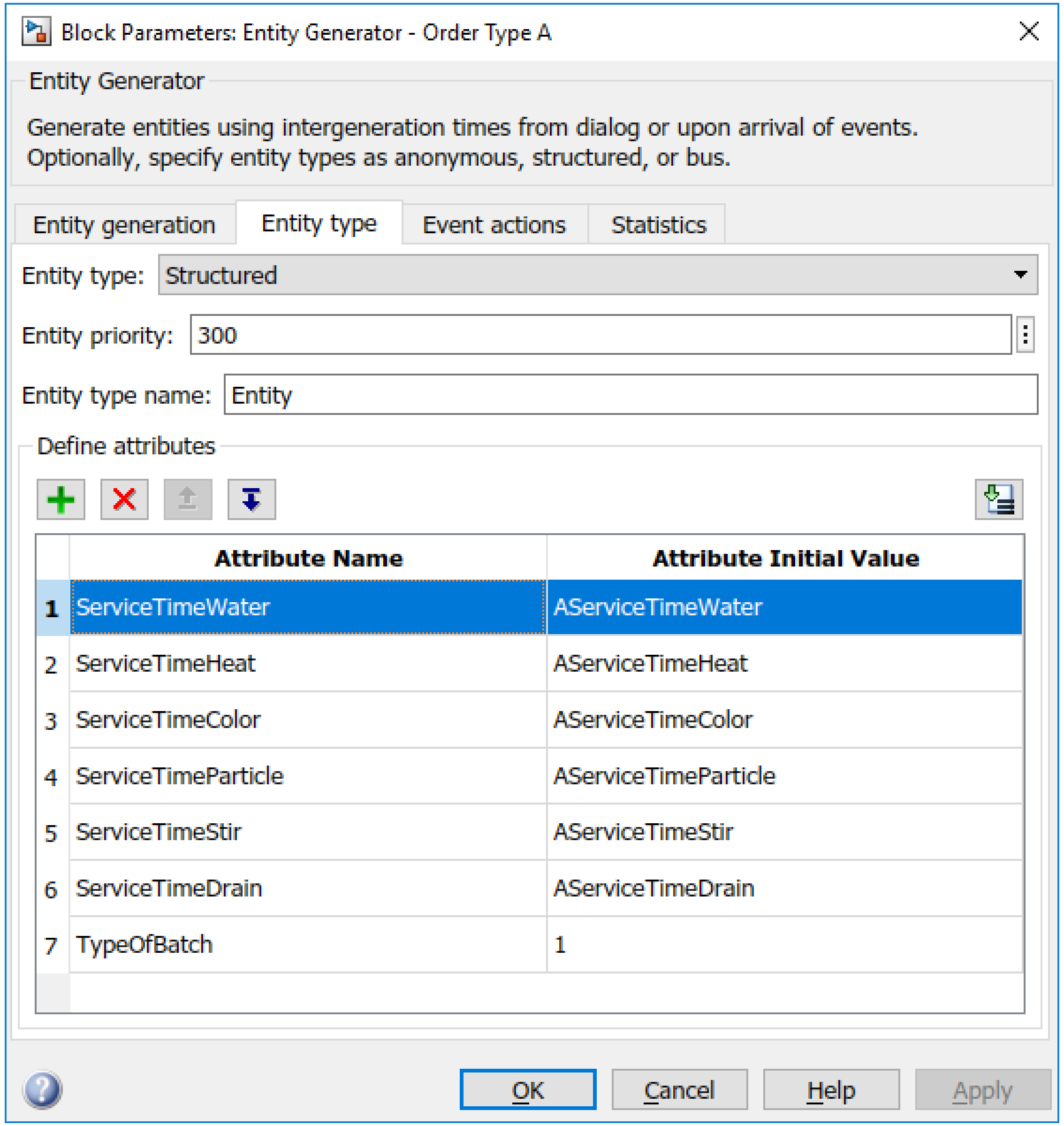


Order (Entities)

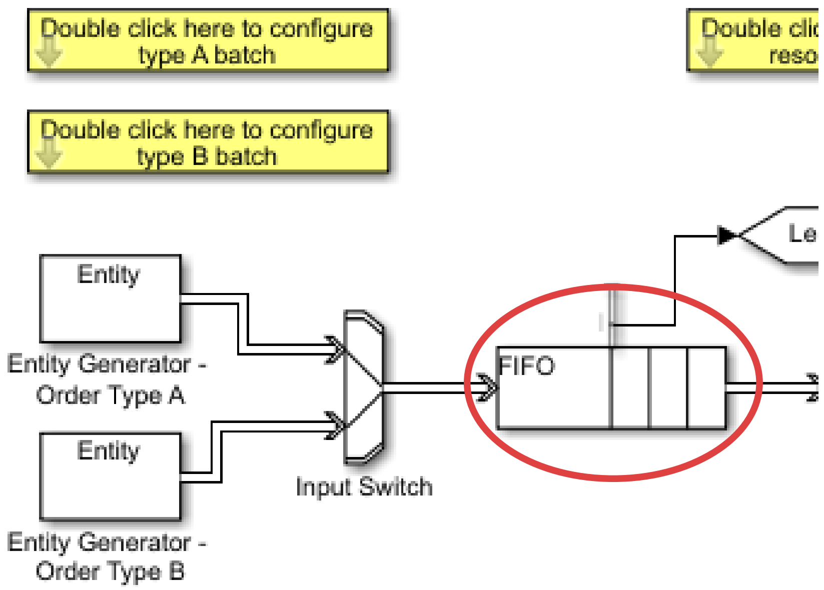


Order (Entities)

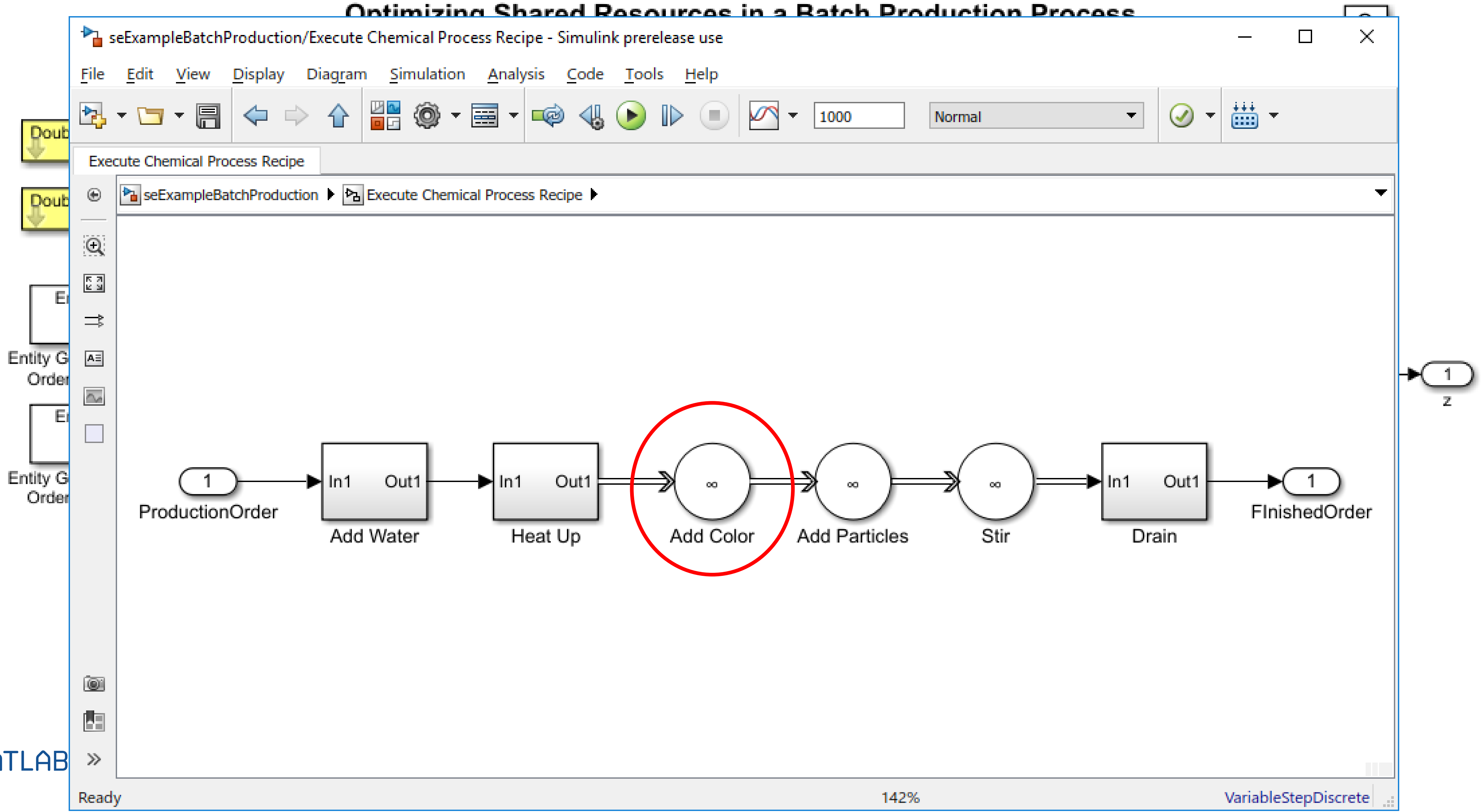




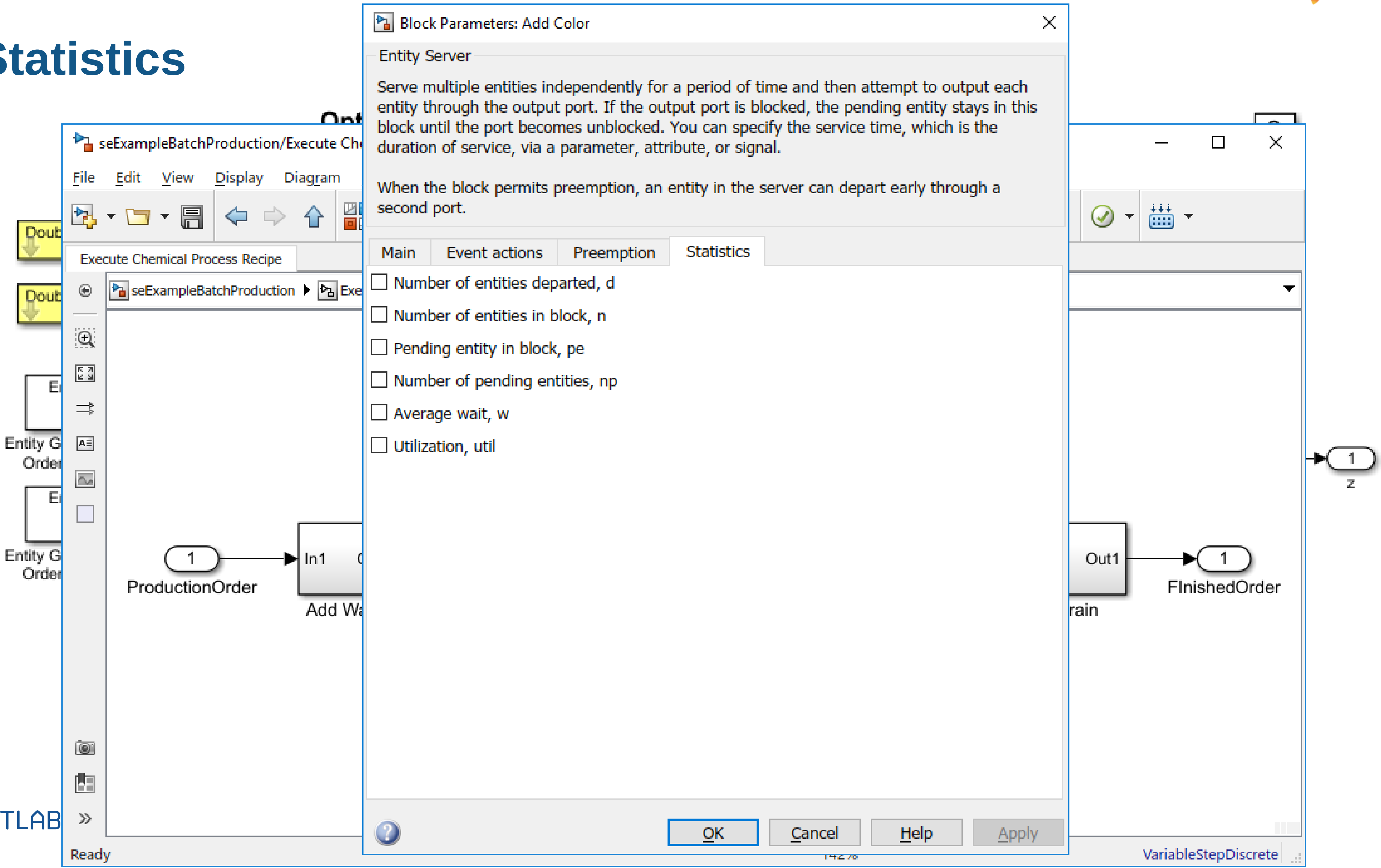
Order queue (Queue)



Processing Unit (Server)



Statistics

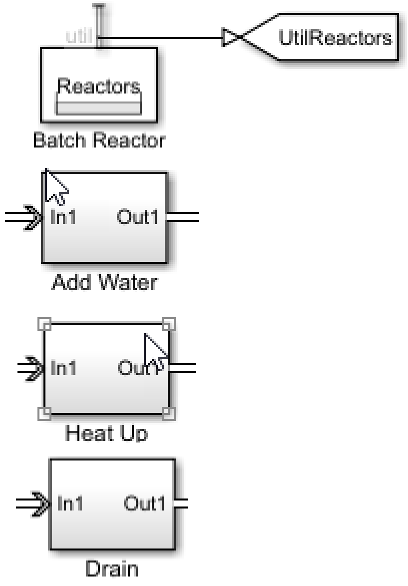


Block Parameters: Add Color
Entity Server
Serve multiple entities independently for a period of time and then attempt to output each entity through the output port. If the output port is blocked, the pending entity stays in this block until the port becomes unblocked. You can specify the service time, which is the duration of service, via a parameter, attribute, or signal.
When the block permits preemption, an entity in the server can depart early through a second port.
Main
Event actions
Preemption
Statistics
Number of entities departed, d
Number of entities in block, n
Pending entity in block, pe
Number of pending entities, np
Average wait, w
Utilization, util
OK
Cancel
Help
Apply
seExampleBatchProduction/Execute Che
File
Edit
View
Display
Diagram
Execute Chemical Process Recipe
seExampleBatchProduction
ProductionOrder
In1
Add Wa
Out1
FInishedOrder
Ready
VariableStepDiscrete

Optimization Model

Decision variables: Number of each type of resource:

- batch reactors
- water tanks
- heaters
- drains



Objective: Minimize weighted sum of the backlog as computed by the simulation model and the cost of the resources

Constraints: Resources are integral; upper and lower limits

Solver: Genetic Algorithm

Optimization Model

- Group the decision variables into an array

```
% Decision variables are the
%     # of batch reactors
%     # of water tanks
%     # of heaters
%     # of drains
```

- Specify upper and lower bounds on the decision variables

```
% Lower bound of decision variables
lb = [1 1 1 1];

% Upper bound of decision variables
ub = [20 10 10 10];
```

- Specify that the decision variables must have integer values

```
% Integer constraints
IntCon = [1 2 3 4];
```

Optimization Model

- Write a function that calls the simulation in order to compute the objective function value

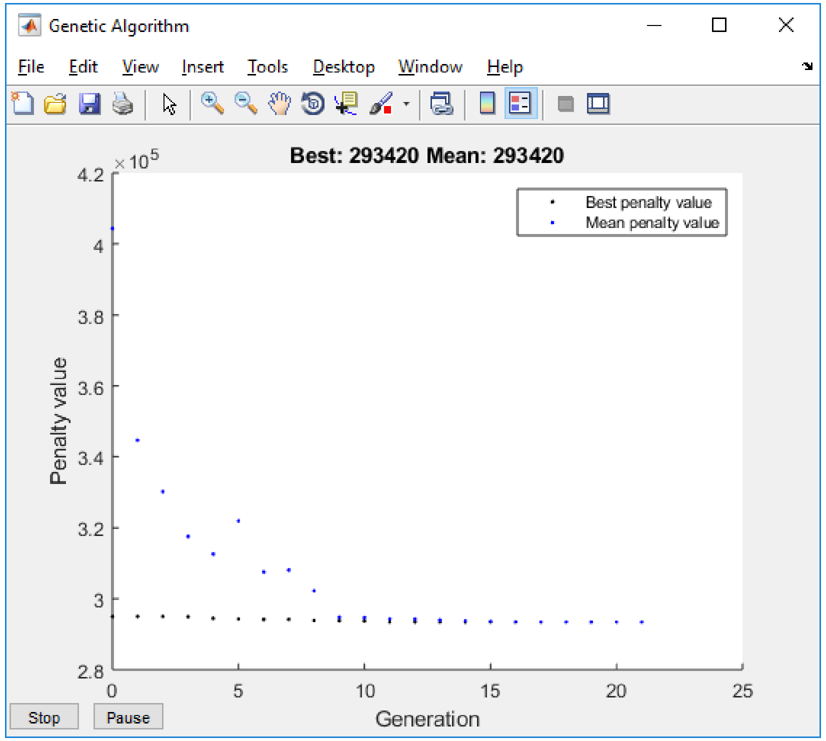
```
% Cost function that assign different values to the decision variables in
% the model
function obj = productionCost(ResourceCapacity)
% Assigns costs to the values of ResourceCapacity, which correspond
% to [batch reactors, water tanks, heaters, drains]
cost = [1000 300 200 100] * ResourceCapacity';

% Assigns variables to the base workspace for simulation
assignin('base', 'ResourceCapacity', ResourceCapacity);

% Simulation of the model and assigns output to the variable z
if isempty(find_system('type', 'block_diagram', ...
        'Name', 'seExampleBatchProduction'))
    load_system('seExampleBatchProduction');
end

set_param('seExampleBatchProduction/ConfigResource','NumBatchReactor', ...
    num2str(ResourceCapacity(1)), 'NumWater',...
    num2str(ResourceCapacity(2)), 'NumHeat', ...
    num2str(ResourceCapacity(3)), 'NumDrain', ...
    num2str(ResourceCapacity(4)));

[~, ~, z] = sim('seExampleBatchProduction');

% Takes the last value of the logged data as the final backlog
% value
backlog = z(end);

% Calculates the objective function, based on the backlog and costs
obj = backlog*10000 + cost;

end
```

Optimization Model

- Solve

```
% Execute genetic algorithm solver
[finalResult, ~, ~] = ga(@productionCost, 4, [], [], [], [], ...
    lb, ub, [], IntCon, opts);
```

```
>> finalResult
finalResult =
    20     1     1     1
```



Conclusion

Steps in Optimization Modeling

- Get overall idea of the system
- What is the goal? What are you trying to achieve?
- Identify variables
- Identify constraints
- Identify the inputs and outputs you can control
- Specify all quantities mathematically
- Check the model for completeness and correctness

Getting started with Optimization using MATLAB

- Optimization examples:
  - https://www.mathworks.com/help/releases/R2018a/optim/examples.html
- Optimization Decision table:
  - https://www.mathworks.com/help/releases/R2018a/optim/ug/optimization-decision-table.html
- Discrete Event Simulation examples:
  - https://www.mathworks.com/help/releases/R2018a/simevents/examples.html

MathWorks Training Offerings

MATLAB Fundamentals

This three-day course provides a comprehensive introduction to the MATLAB® technical computing environment. No prior programming experience or knowledge of MATLAB is assumed. Themes of data analysis, visualization, modeling, and programming are explored throughout the course. Topics include:

- Working with the MATLAB user interface
- Entering commands and creating variables
- Analyzing vectors and matrices
- Visualizing vector and matrix data
- Working with data files
- Working with data types
- Automating commands with scripts
- Writing programs with branching and loops
- Writing functions

MathWorks Training Offerings

Optimization Techniques using MATLAB

This one-day course introduces applied optimization in the MATLAB® environment, focusing on using Optimization Toolbox™ and Global Optimization Toolbox™. Topics include:

- Running optimization problems in MATLAB
- Specifying objective functions
- Specifying constraints
- Choosing solvers and algorithms
- Evaluating results and improving performance
- Using global optimization methods

MathWorks Training Offerings

SimEvents for Discrete-Event System Modeling

This one-day course focuses on modeling event-driven systems in Simulink® using SimEvents®. Topics include:

- Creating discrete-event models
- Defining attributes and event actions
- Controlling queue and server behavior
- Developing variable model topologies using routing and resources
- Integrating discrete-event and time-domain systems
- Determining optimal system parameters

Contact us

Speakers' Details

Email:

Gautam.PalachandaChengappa@mathworks.in

Viju.Ravichandran@mathworks.in

LinkedIn:

https://in.linkedin.com/in/gautamponnappa

https://in.linkedin.com/in/lravichandran

Contact MathWorks India

Products/Training Enquiry Booth

Call: 080-6632-6000

Email: info@mathworks.in

- **Share your session feedback:**

  Please fill in your feedback for this session in the feedback form