A Unified Machine Learning Framework for Optimization Under Uncertainty

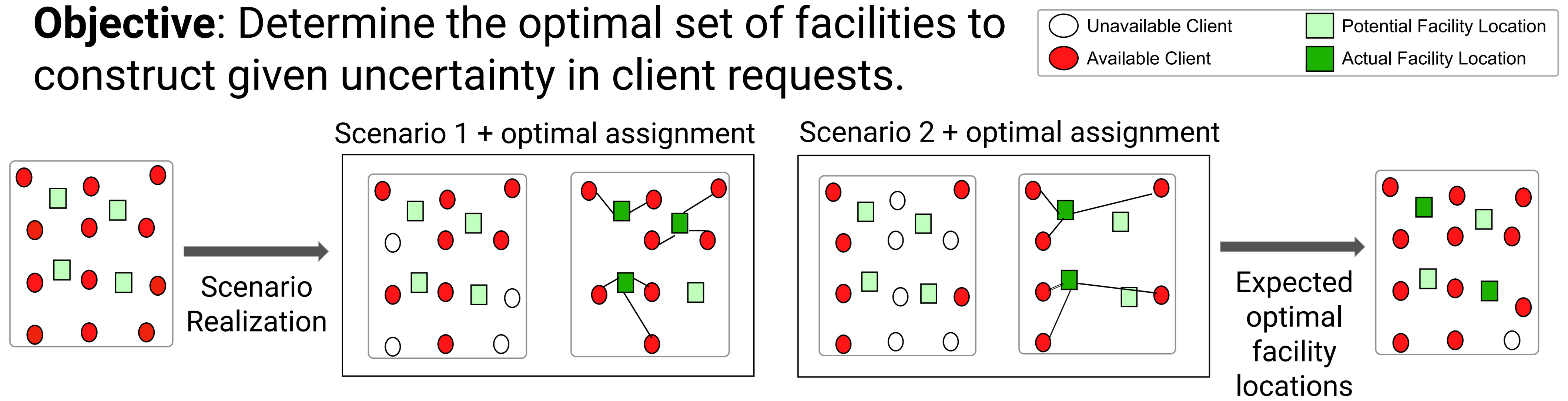
Justin Dumouchelle

Learning for **Two-Stage Stochastic Programming (2SP)**

Published at NeurIPS 2022. Joint work with Rahul Patel, Elias Khalil, and Merve Bodur

Stochastic Facility Location Problem

Objective: Determine the optimal set of facilities to construct given uncertainty in client requests.

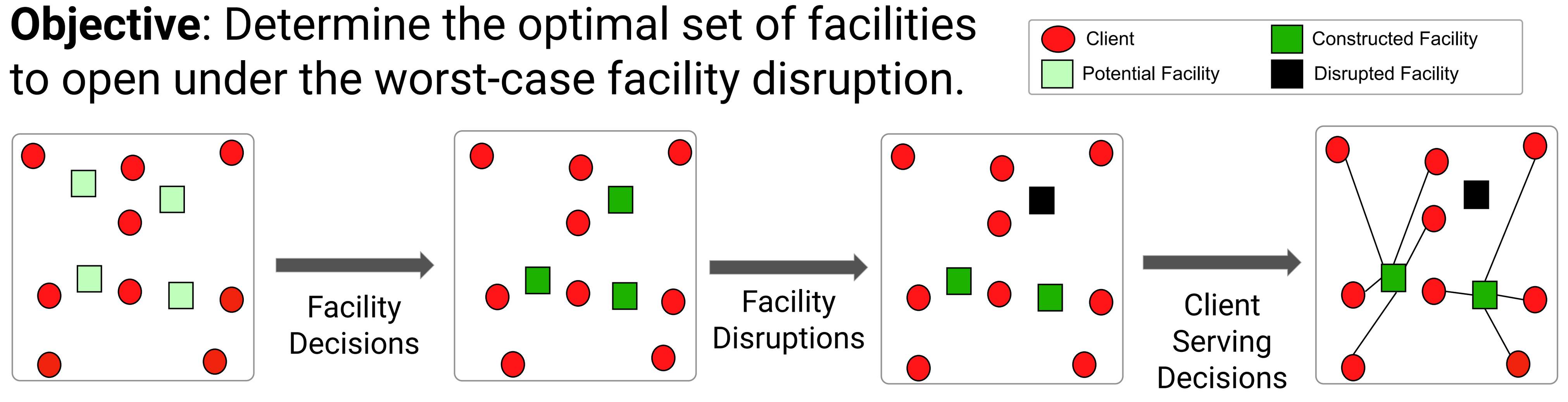


Learning for **Two-Stage Robust Optimization (2RO)**

Under review. Joint work with Esther Julien, Jannis Kurtz, and Elias Khalil

Robust Facility Location Problem

Objective: Determine the optimal set of facilities to open under the worst-case facility disruption.



Overall framework

Data Collection
- Sample decisions + uncertainty.
- Solve subproblems with off-the-shelf solvers.

GUROBI OPTIMIZATION

Supervised Learning
- Train an NN using off-the-shelf ML packages.

PyTorch

Evaluation on new instances
- Solve the surrogate MILP with an off-the-shelf solver.

GUROBI OPTIMIZATION

Optimization Formulation

Objective: Determine optimal first-stage decisions that minimize sum of the first-stage cost and expected second-stage cost.

Challenge: Exact optimization becomes exponentially harder with the number of observed scenarios, K. Integer decision further aggravates intractability.

First-Stage Problem

$$\min_{\underbrace{\mathbf{x}}_{\text{First-stage decisions}}} \Big\{ \overbrace{\mathbf{c}^\intercal\mathbf{x}}^{\text{First-stage cost}} + \underbrace{\sum_{k=1}^{K} p_k Q(\mathbf{x},\xi_k)}_{\text{Expected second-stage cost}} : \mathbf{x}\in\mathcal{X} \Big\}$$

Second-Stage Problem

$$Q(\mathbf{x},\xi) := \min_{\underbrace{\mathbf{y}}_{\text{Second-stage decisions}}} \Big\{ \overbrace{F(\mathbf{y},\xi)}^{\text{Second-stage cost}} : \mathbf{y}\in\mathcal{Y}(\mathbf{x},\xi) \Big\}$$

Objective: Determine optimal first-stage decisions that minimize the sum of the first-stage cost and worst-case cost.

Challenge: Solving the nested optimization problem is intractable and specialized algorithms only exists for a limited classes of problems.

First-Stage Problem

$$\min_{\underbrace{\mathbf{x}}_{\text{First-stage decisions}}} \Big\{ \overbrace{\mathbf{c}^\intercal\mathbf{x}}^{\text{First-stage cost}} + \underbrace{Q(\mathbf{x})}_{\text{Worst-case cost}} : \mathbf{x}\in\mathcal{X} \Big\}$$

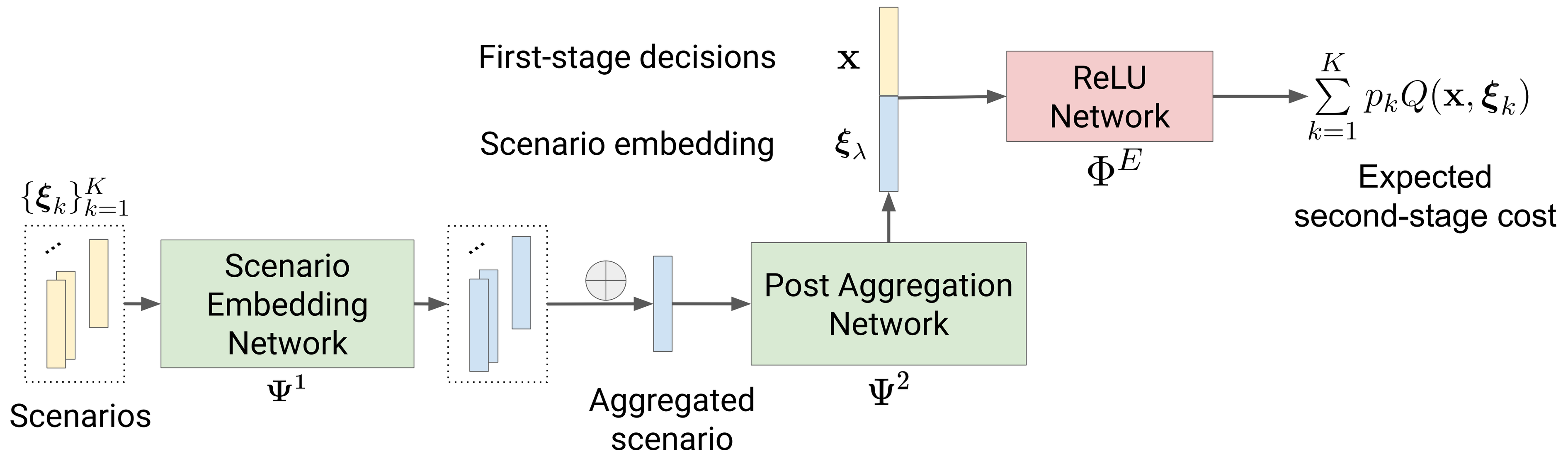
Second-Stage Problem

$$Q(\mathbf{x}) := \max_{\underbrace{\xi\in\Xi}_{\text{Worst-case uncertainty}}} \min_{\underbrace{\mathbf{y}}_{\text{Second-stage decisions}}} \Big\{ \overbrace{\mathbf{d}(\xi)^\intercal\mathbf{y}}^{\text{Second-stage cost}} : \mathbf{y}\in\mathcal{Y}(\mathbf{x},\xi) \Big\}$$

Machine Learning Methodology

ML Solution: Replace the expected stage-cost value function with a neural network approximation.

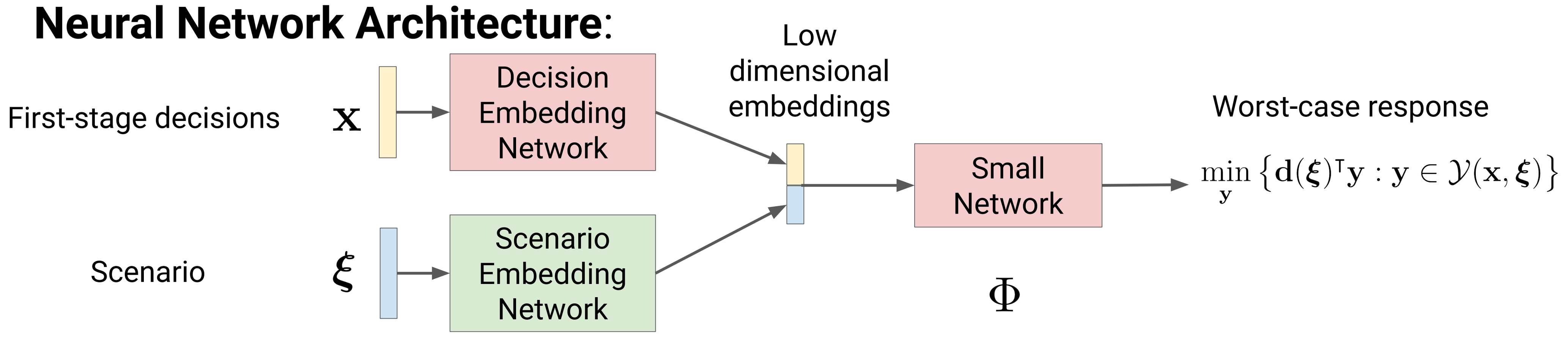
Neural Network Architecture:



Surrogate Optimization Model: $\min_{\mathbf{x}} \{ \mathbf{c}^\intercal\mathbf{x} + \mathrm{NN}(\mathbf{x}, \{\xi_k\}_{k=1}^K) : \mathbf{x}\in\mathcal{X} \}$

ML Solution: Replace the worst-case response optimization problem with a neural network approximation.

Neural Network Architecture:



Surrogate Optimization Model: $\min_{\mathbf{x},\mathbf{y},\xi_a} \{ \mathbf{c}^\intercal\mathbf{x} + \mathbf{d}(\xi_a) : \mathbf{x}\in\mathcal{X}, \mathbf{y}\in\mathcal{Y}(\mathbf{x},\xi_a), \xi_a \in \arg\max_{\xi\in\Xi'} \{\mathrm{NN}(\mathbf{x},\xi)\} \}$

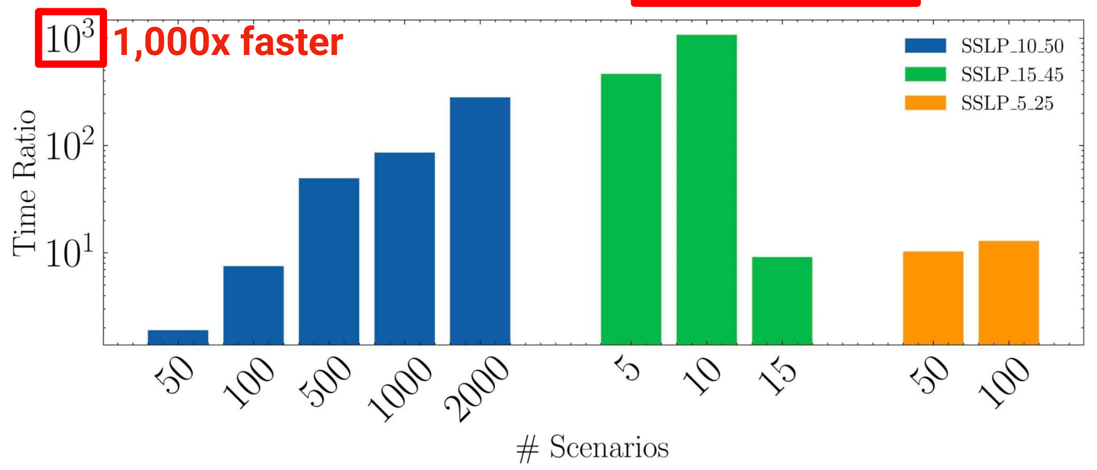
- Optimization problem solved via constraint generation of small networks.
- Worst-case scenarios determined via adversarial problem ($\max_{\xi\in\Xi} \mathrm{NN}(\hat{\mathbf{x}},\xi)$).

Experimental Results

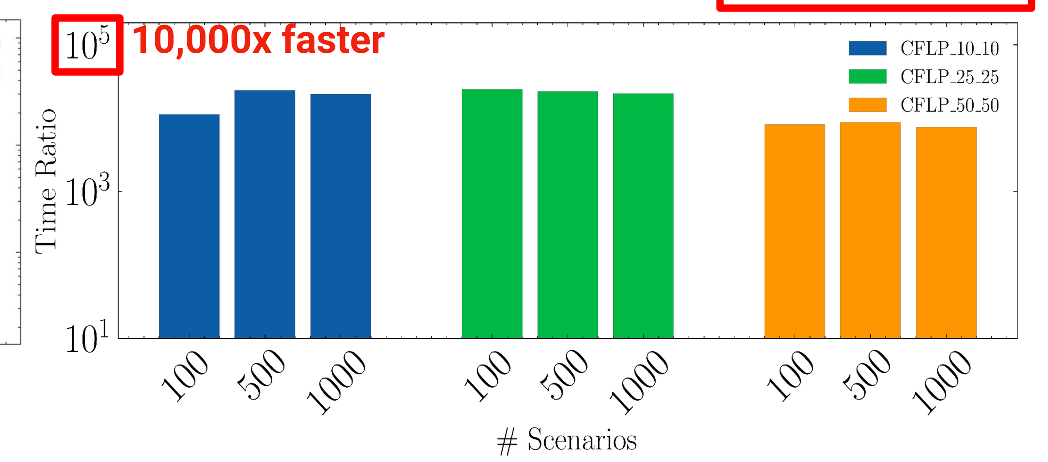
Stochastic Programming

Reduction factor in solving time over baseline (Integer L-shaped/EF) (higher is better).

Stochastic Server Location (Gap: 0.87%)



Capacitated Facility Location (Gap: -2.93%)

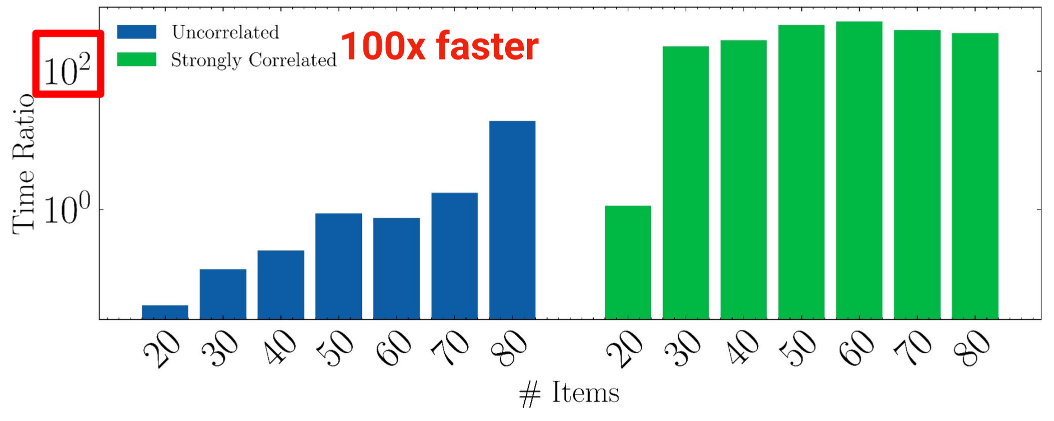


- ML finds OPT on most SSLP instances and best known solutions on most CFLP instances.

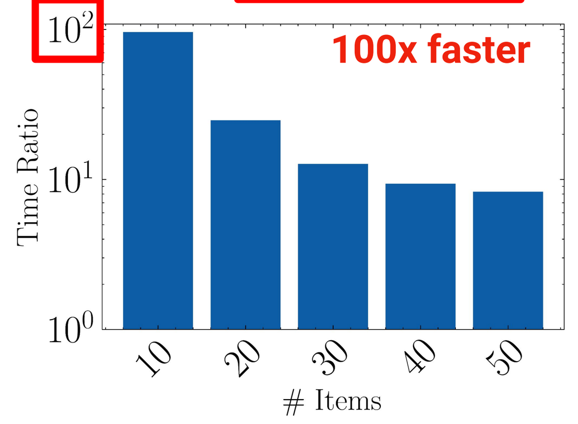
Robust Optimization

Reduction factor in solving time over baseline (branch-and-price/k-adaptability) (higher is better).

Robust Knapsack Problem (Gap: 1.30%)



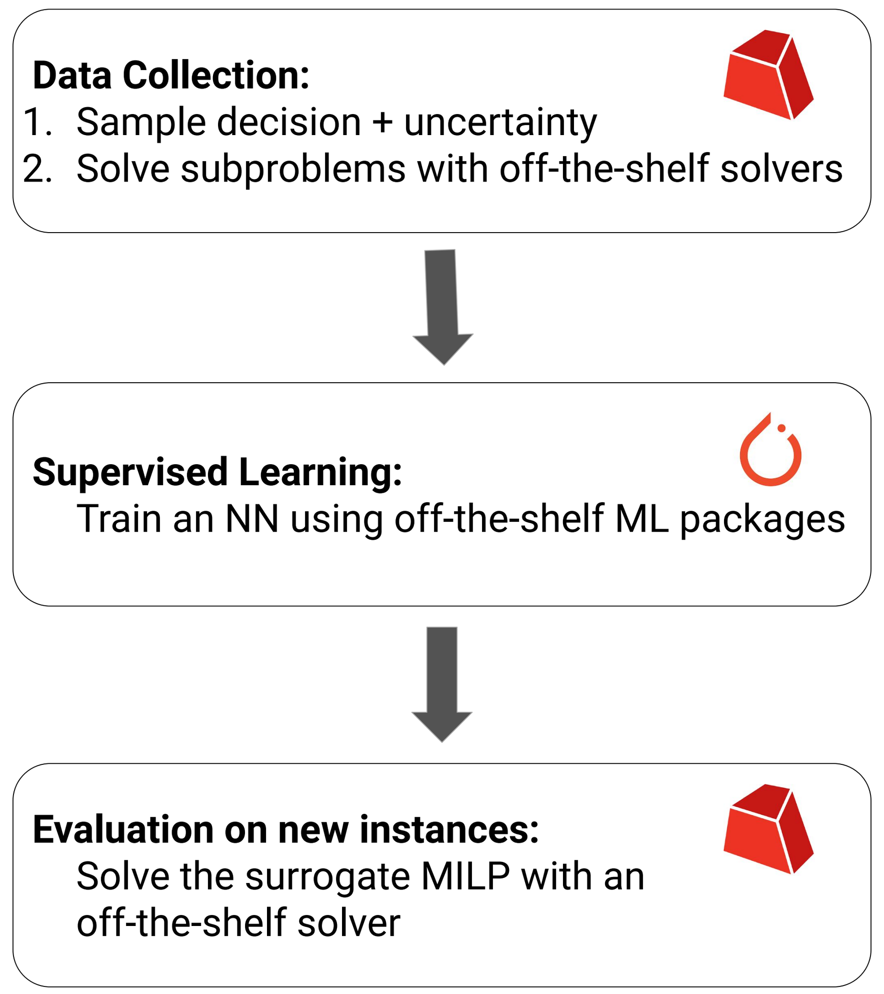
Capital Budgeting (Gap: 0.33%)

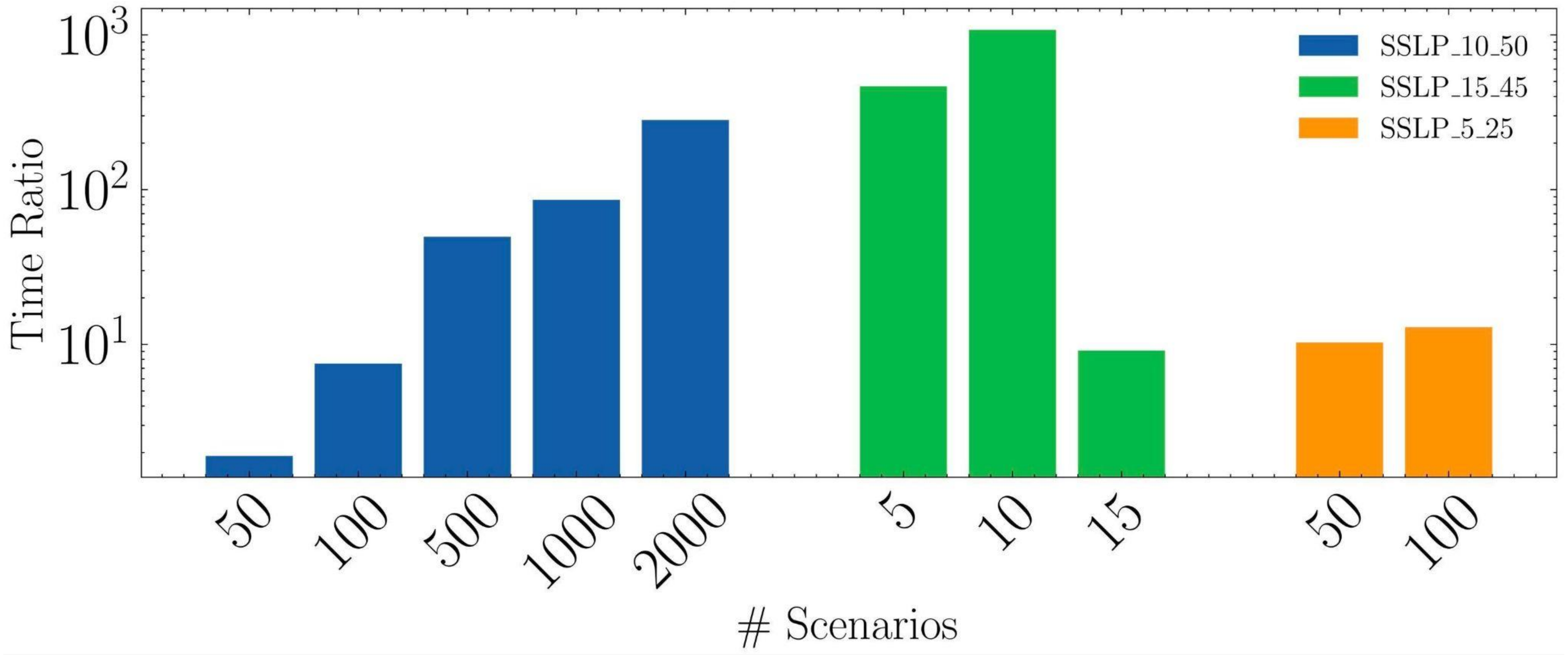


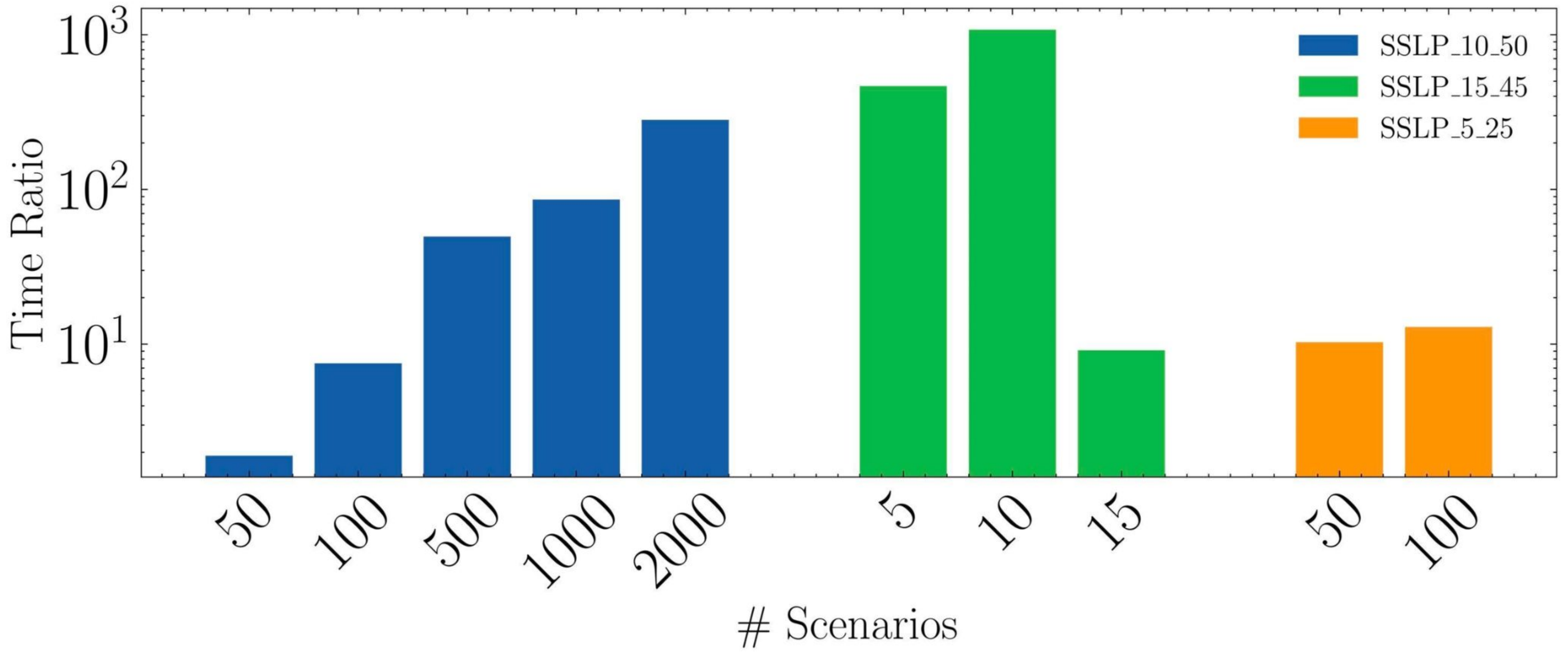
- ML finds best solutions on difficult instances; near OPT on easy instances.

2SP **2RO**

Mechanical & Industrial Engineering
UNIVERSITY OF TORONTO







(these can be presented visually using a flowchart and logos for Gurobi/PyTorch to make it more concrete)

1. Data collection using off-the-shelf solvers
2. Supervised training of NN using off-the-shelf ML packages
3. Given a new instance: solving the surrogate MILP with an off-the-shelf solver

A Unified Machine Learning Framework for Optimization Under Uncertainty

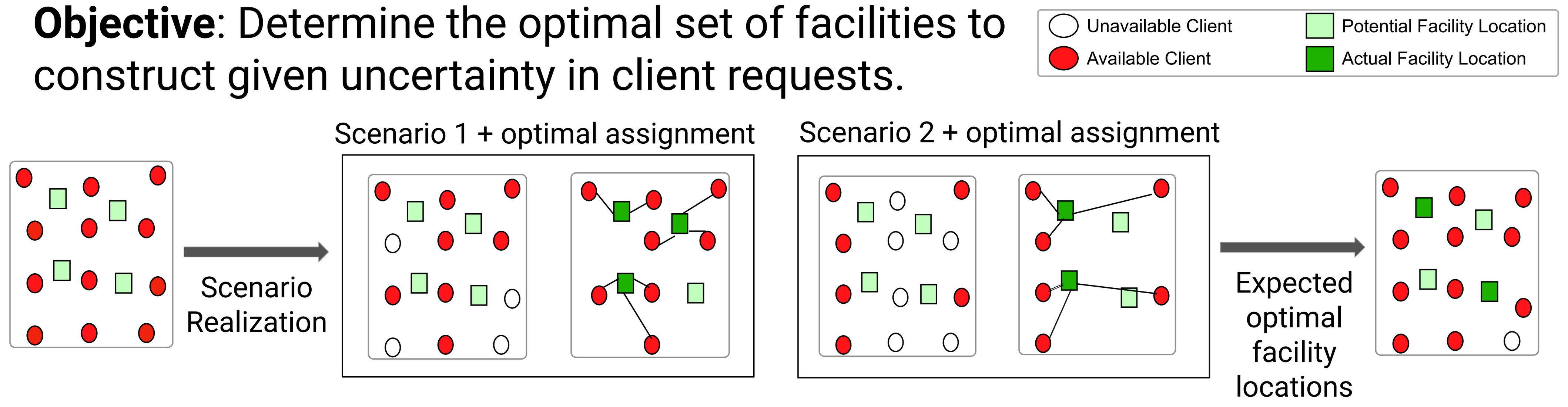
Justin Dumouchelle

Learning for **Two-Stage Stochastic Programming (2SP)**

Published at NeurIPS 2022. Joint work with Rahul Patel, Elias Khalil, and Merve Bodur

Stochastic Facility Location Problem

Objective: Determine the optimal set of facilities to construct given uncertainty in client requests.

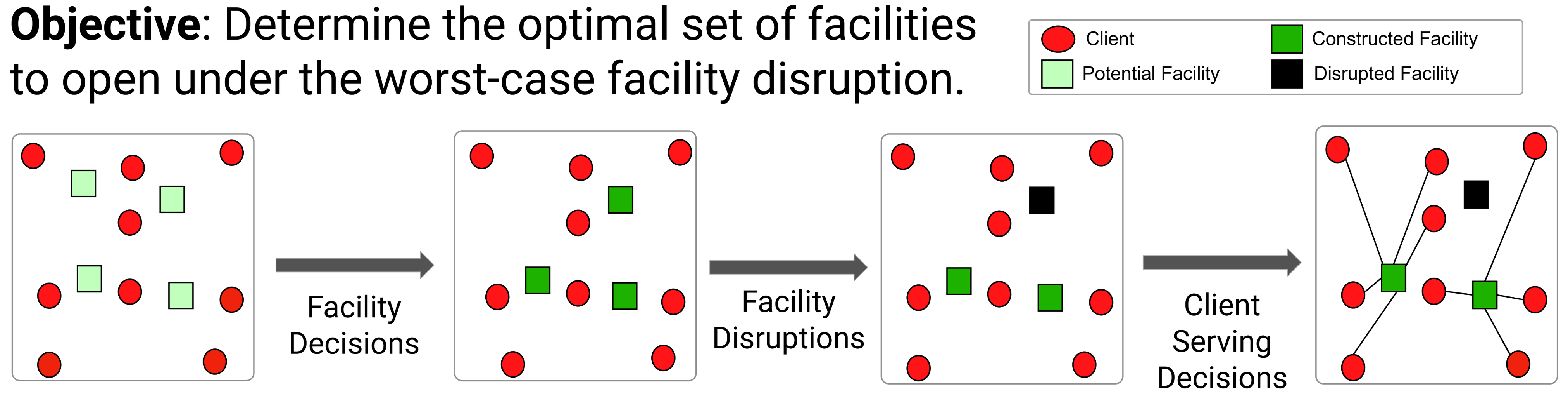


Learning for **Two-Stage Robust Optimization (2RO)**

Under review. Joint work with Esther Julien, Jannis Kurtz, and Elias Khalil

Robust Facility Location Problem

Objective: Determine the optimal set of facilities to open under the worst-case facility disruption.



Overall framework

Data Collection
- Sample decisions + uncertainty.
- Solve subproblems with off-the-shelf solvers.

Supervised Learning
- Train an NN using off-the-shelf ML packages.

Evaluation on new instances
- Solve the surrogate MILP with an off-the-shelf solver.

Optimization Formulation

Objective: Determine optimal first-stage decisions that minimize sum of the first-stage cost and expected second-stage cost.

Challenge: Exact optimization becomes exponentially harder with the number of observed scenarios, K. Integer decision further aggravates intractability.

First-Stage Problem

$$\min_{\mathbf{x}} \Big\{ \underbrace{\mathbf{c}^\intercal \mathbf{x}}_{\text{First-stage cost}} + \underbrace{\sum_{k=1}^{K} p_k Q(\mathbf{x}, \xi_k)}_{\text{Expected second-stage cost}} : \mathbf{x} \in \mathcal{X} \Big\}$$

First-stage decisions: $\mathbf{x}$

Second-Stage Problem

$$Q(\mathbf{x}, \xi) := \min_{\mathbf{y}} \Big\{ \underbrace{F(\mathbf{y}, \xi)}_{\text{Second-stage cost}} : \mathbf{y} \in \mathcal{Y}(\mathbf{x}, \xi) \Big\}$$

Second-stage decisions: $\mathbf{y}$

Objective: Determine optimal first-stage decisions that minimize the sum of the first-stage cost and worst-case cost.

Challenge: Solving the nested optimization problem is intractable and specialized algorithms only exists for a limited classes of problems.

First-Stage Problem

$$\min_{\mathbf{x}} \Big\{ \underbrace{\mathbf{c}^\intercal \mathbf{x}}_{\text{First-stage cost}} + \underbrace{Q(\mathbf{x})}_{\text{Worst-case cost}} \Big) : \mathbf{x} \in \mathcal{X} \Big\}$$

First-stage decisions: $\mathbf{x}$

Second-Stage Problem

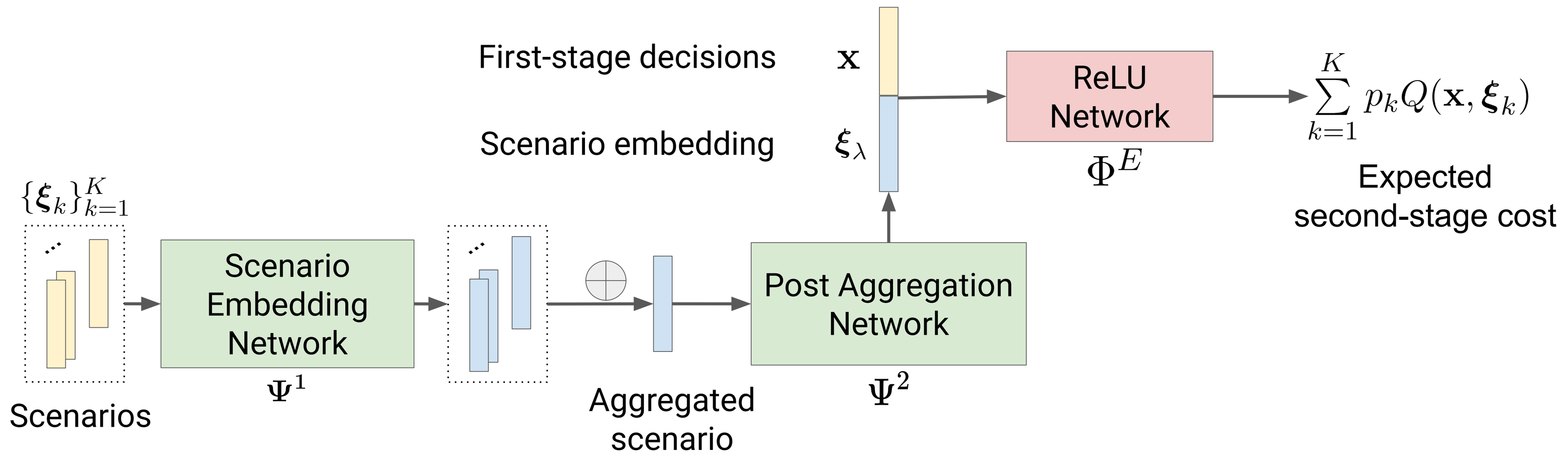
$$Q(\mathbf{x}) := \max_{\xi \in \Xi} \min_{\mathbf{y}} \Big\{ \underbrace{\mathbf{d}(\xi)^\intercal \mathbf{y}}_{\text{Second-stage cost}} : \mathbf{y} \in \mathcal{Y}(\mathbf{x}, \xi) \Big\}$$

Worst-case uncertainty: $\xi \in \Xi$; Second-stage decisions: $\mathbf{y}$

Machine Learning Methodology

ML Solution: Replace the expected stage-cost value function with a neural network approximation.

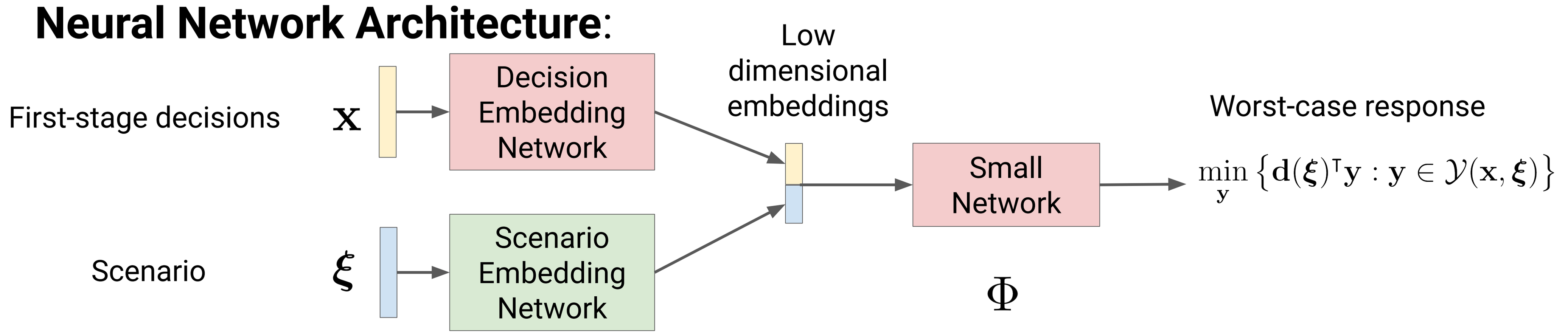
Neural Network Architecture:



Surrogate Optimization Model: $\min_{\mathbf{x}} \{ \mathbf{c}^\intercal \mathbf{x} + \mathrm{NN}(\mathbf{x}, \{\xi_k\}_{k=1}^K) : \mathbf{x} \in \mathcal{X} \}$

ML Solution: Replace the worst-case response optimization problem with a neural network approximation.

Neural Network Architecture:



Surrogate Optimization Model: $\min_{\mathbf{x}, \mathbf{y}, \xi_a} \{ \mathbf{c}^\intercal \mathbf{x} + \mathbf{d}(\xi_a) : \mathbf{x} \in \mathcal{X}, \mathbf{y} \in \mathcal{Y}(\mathbf{x}, \xi_a), \xi_a \in \arg\max_{\xi \in \Xi'} \{ \mathrm{NN}(\mathbf{x}, \xi) \} \}$

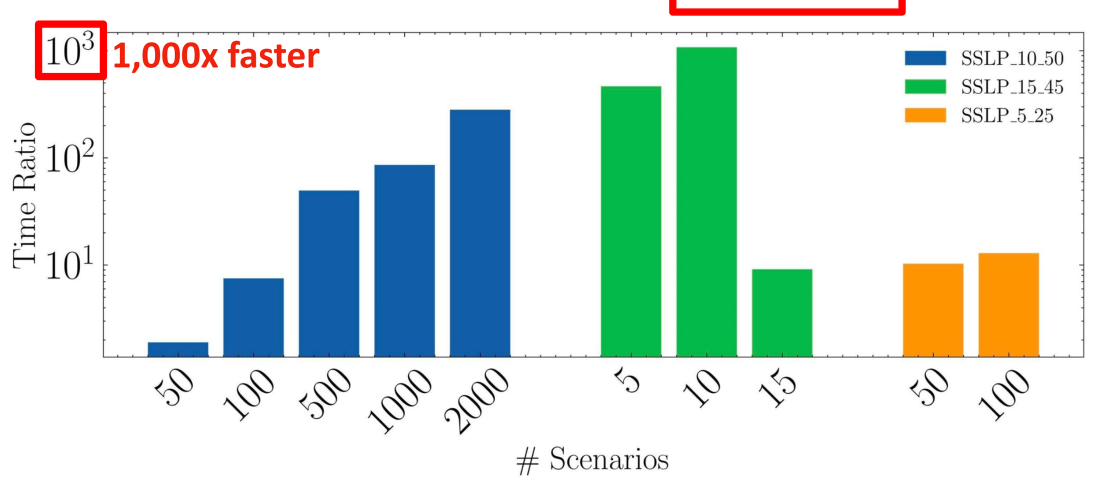
- Optimization problem solved via constraint generation of small networks.
- Worst-case scenarios determined via adversarial problem ($\max_{\xi \in \Xi} \mathrm{NN}(\hat{\mathbf{x}}, \xi)$).

Experimental Results

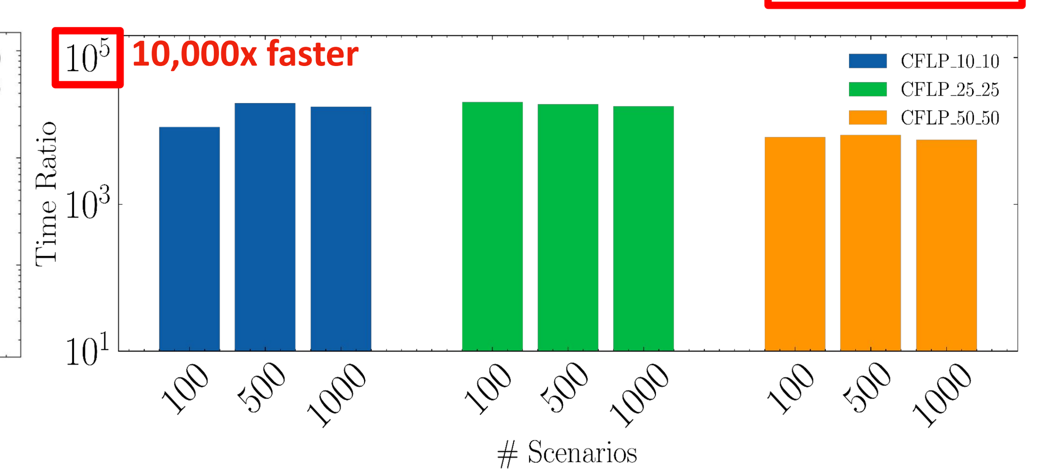
Stochastic Programming

Reduction factor in computing time over baseline (Integer L-shaped/EF) (higher is better).

Stochastic Server Location Problem (Gap: 0.87%)



Capacitated Facility Location Problem (Gap: -2.93%)

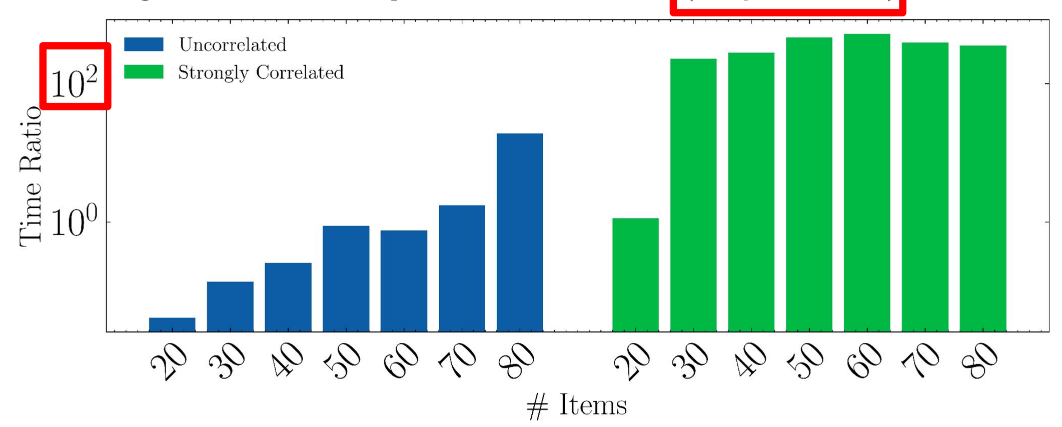


- ML finds OPT on most instances.
- ML finds best solutions on most instances.

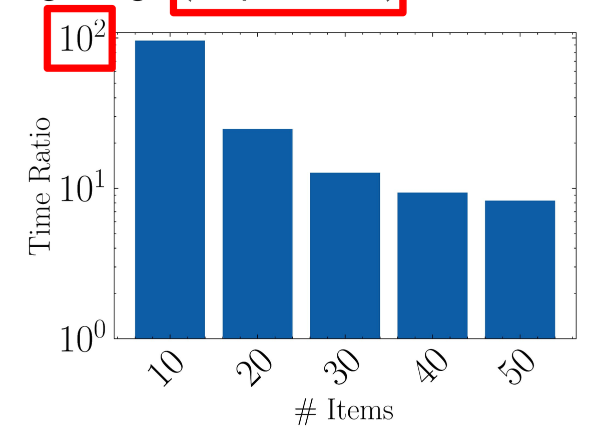
Robust Optimization

Reduction factor in computing time over baseline (branch-and-price/k-adaptability) (higher is better).

Two-Stage Robust Knapsack Problem (Gap: 1.30%)



Capital Budgeting (Gap: 0.33%)



- ML finds best solutions on difficult instances; near optimal on easy instances.
- Time reductions on the order of 10-100x the state-of-the-art baselines.

Mechanical & Industrial Engineering
UNIVERSITY OF TORONTO

A Unified Machine Learning Framework for Optimization Under Uncertainty

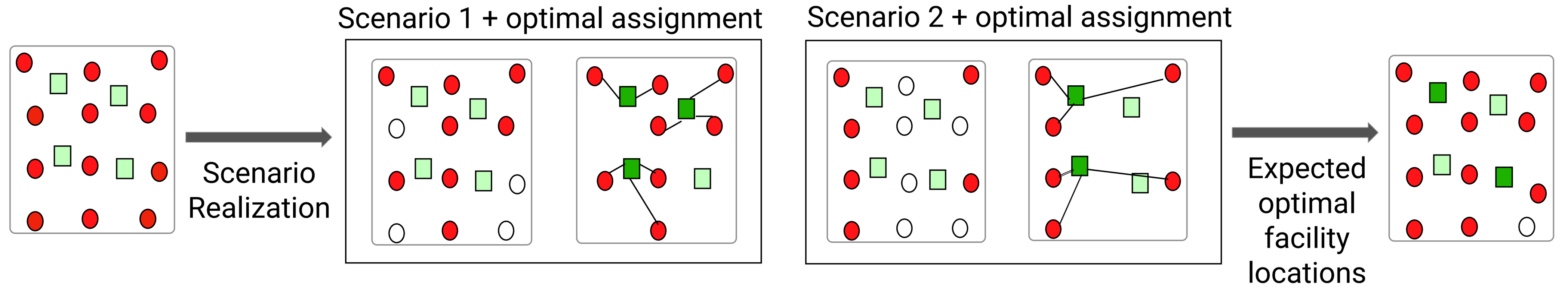
Justin Dumouchelle

Learning for **Two-Stage Stochastic Programming (2SP)**

Published at NeurIPS 2022. Joint work Rahul Patel, Elias Khalil, and Merve Bodur

Stochastic Facility Location Problem

Objective: Determine the optimal set of facilities to construct given uncertainty in client requests.

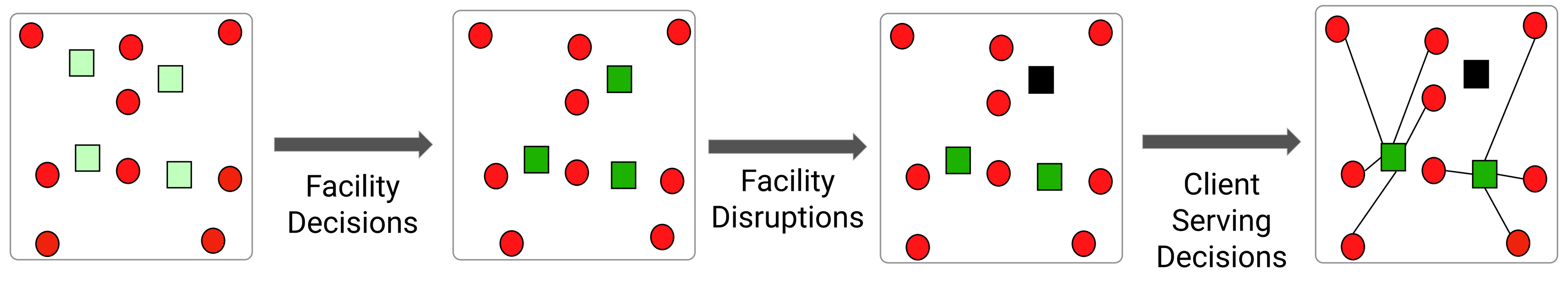


Learning for **Adjustable Robust Optimization (2RO)**

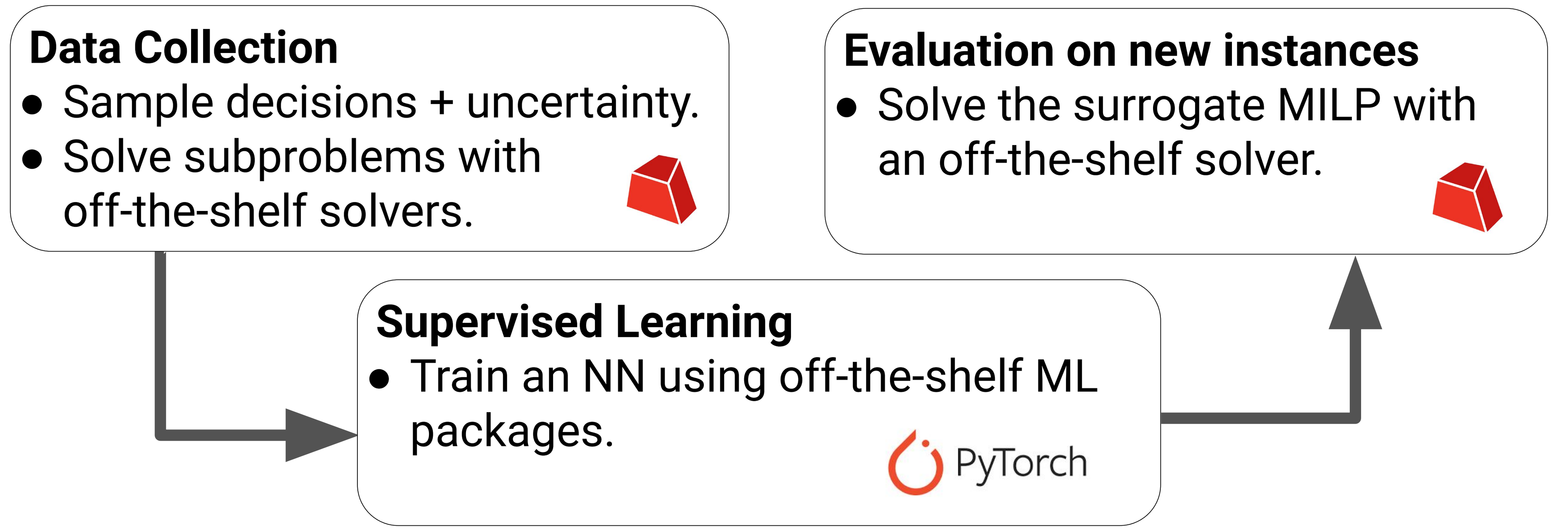
Under review

Robust Facility Location Problem

Objective: Determine the optimal set of facilities to open under the worst-case facility disruption.



Overall framework



Optimization Formulation

Objective: Determine optimal first-stage decisions that minimize sum of the first-stage cost and expected second-stage cost.

Challenge: Exact optimization becomes exponentially harder with the number of observed scenarios, K. Integer decision further aggravates intractability.

First-Stage Problem

$$\min_{\underbrace{\mathbf{x}}_{\text{First-stage decisions}}} \Big\{\underbrace{\mathbf{c}^\intercal\mathbf{x}}_{\text{First-stage cost}} + \underbrace{\sum_{k=1}^{K} p_k Q(\mathbf{x},\xi_k)}_{\text{Expected second-stage cost}} : \mathbf{x}\in\mathcal{X}\Big\}$$

Second-Stage Problem

$$Q(\mathbf{x},\xi) := \min_{\underbrace{\mathbf{y}}_{\text{Second-stage decisions}}} \Big\{\underbrace{F(\mathbf{y},\xi)}_{\text{Second-stage cost}} : \mathbf{y}\in\mathcal{Y}(\mathbf{x},\xi)\Big\}$$

Objective: Determine optimal first-stage decisions that minimize the sum of the first-stage cost and worst-case cost.

Challenge: Solving the nested optimization problem is intractable and specialized algorithms only exists for a limited classes of problems.

First-Stage Problem

$$\min_{\underbrace{\mathbf{x}}_{\text{First-stage decisions}}} \Big\{\underbrace{\mathbf{c}^\intercal\mathbf{x}}_{\text{First-stage cost}} + \underbrace{Q(\mathbf{x})}_{\text{Worst-case cost}} : \mathbf{x}\in\mathcal{X}\Big\}$$

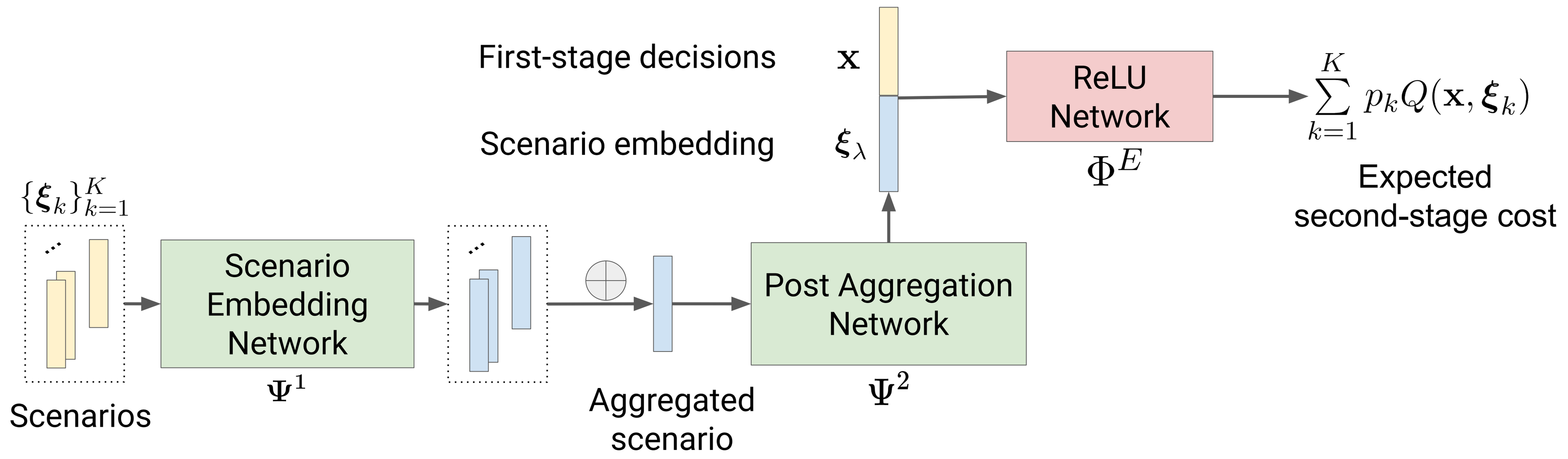
Second-Stage Problem

$$Q(\mathbf{x}) := \max_{\underbrace{\xi\in\Xi}_{\text{Worst-case uncertainty}}} \min_{\underbrace{\mathbf{y}}_{\text{Second-stage decisions}}} \Big\{\underbrace{\mathbf{d}(\xi)^\intercal\mathbf{y}}_{\text{Second-stage cost}} : \mathbf{y}\in\mathcal{Y}(\mathbf{x},\xi)\Big\}$$

Machine Learning Methodology

ML Solution: Replace the expected stage-cost value function with a neural network approximation.

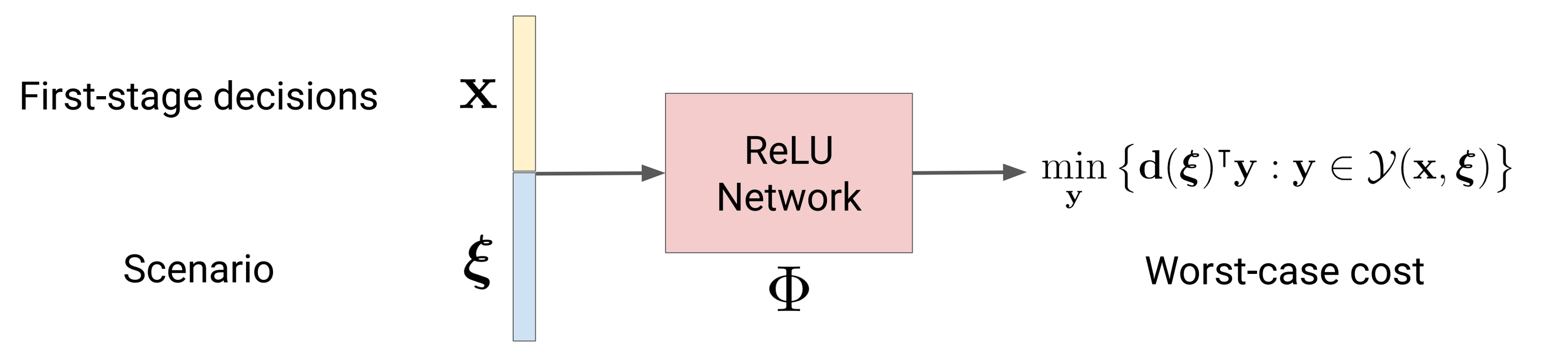
Neural Network Architecture:



Surrogate Optimization Model: $\min_{\mathbf{x}} \{\mathbf{c}^\intercal\mathbf{x} + \mathrm{NN}(\mathbf{x}, \{\xi_k\}_{k=1}^K) : \mathbf{x}\in\mathcal{X}\}$

ML Solution: Replace the worst-case cost with a neural network approximation.

Neural Network Architecture:



Surrogate Optimization Model: $\min_{\mathbf{x}} \{\mathbf{c}^\intercal\mathbf{x} + \eta : \mathbf{x}\in\mathcal{X}, \eta \geq \mathrm{NN}(\mathbf{x},\xi), \forall\xi\in\Xi\}$

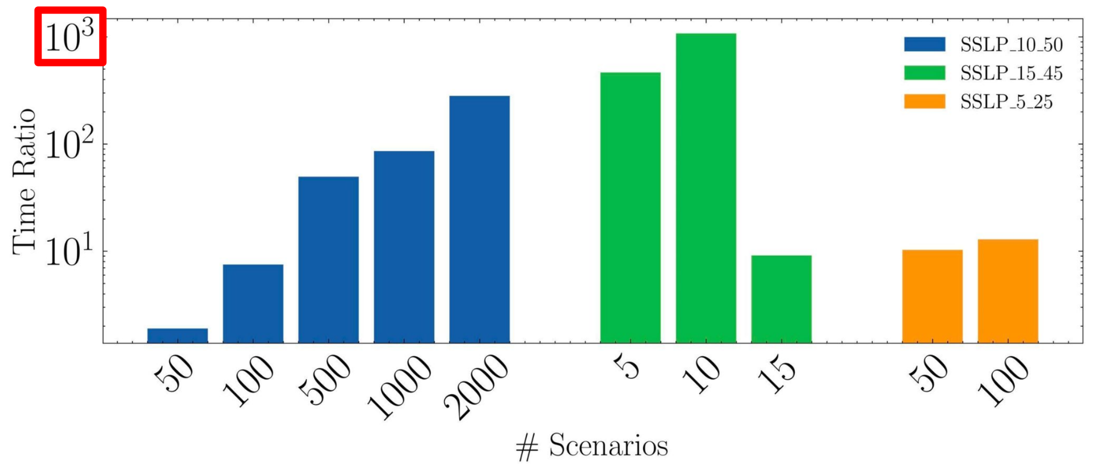
- Optimization problem solved with row generation.
- Worst-case scenarios determined by sampling or adversarial problem.

Experimental Results

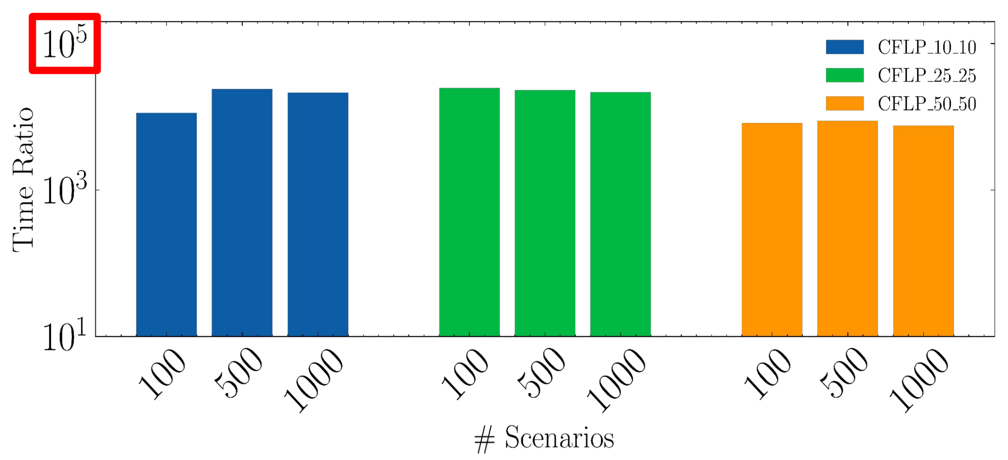
Stochastic Programming

Reduction factor in computing time over baseline (Integer L-shaped/EF) (higher is better).

Stochastic Server Location Problem (Gap: 0.87%)



Capacitated Facility Location Problem (Gap: -2.93%)

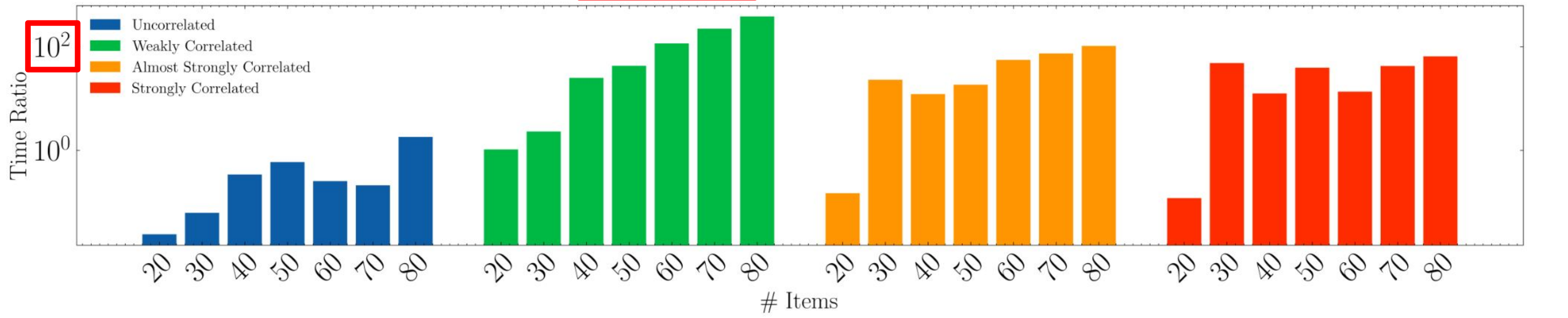


- NN is up to 1,000x faster than baseline.
- Approximation finds optimal solutions on most instances.
- NN is up to 10,000x than faster.
- Approximation finds better solutions than baseline.

Robust Optimization

Reduction factor in computing time over problem specific decomposition (higher is better).

Adjustable Robust Knapsack Problem (Gap: 5.39%)



- Time ratio and relative solution quality improve as problem sizes increase or correlation increases difficult of problem.

Mechanical & Industrial Engineering
UNIVERSITY OF TORONTO

ations

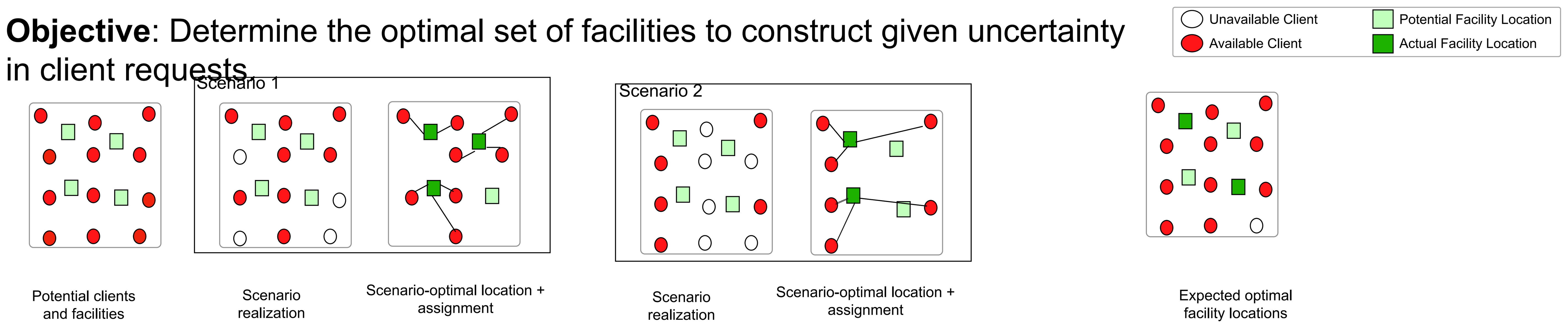
A Unified Machine Learning Framework for Optimization under Uncertainty

Justin Dumouchelle
Mechanical & Industrial Engineering
UNIVERSITY OF TORONTO

Learning for **Two-Stage Stochastic Programming (2SP)**

Stochastic Facility Location Problem

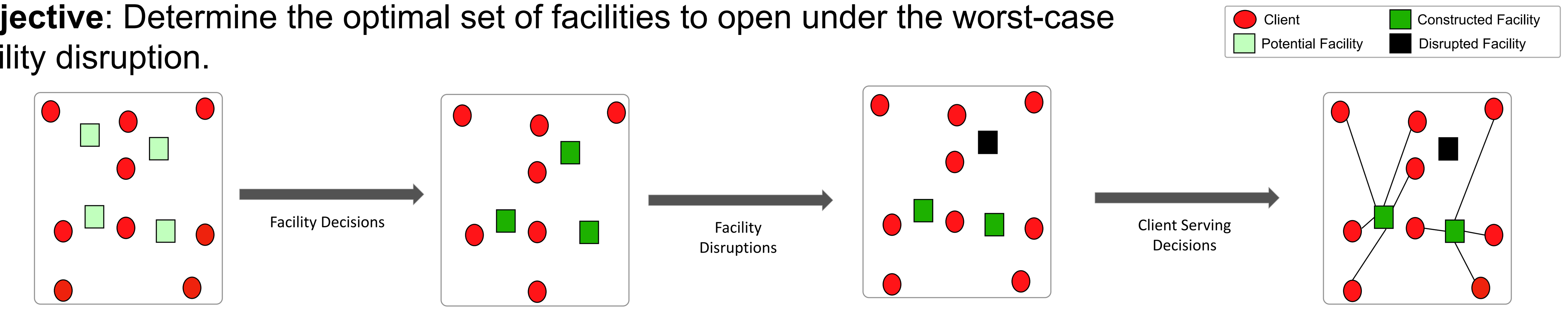
Objective: Determine the optimal set of facilities to construct given uncertainty in client requests.



Learning for **Adjustable Robust Optimization (ARO)**

Robust Facility Location Problem

Objective: Determine the optimal set of facilities to open under the worst-case facility disruption.



Formulation

Objective: Determine optimal first-stage decisions that minimize sum of first-stage cost and expected second-stage cost.
Challenge: Exact optimization becomes exponentially harder with the number of observed scenarios, *K*. Integer decision further aggravates intractability.

First-Stage Problem

$$\min_{\mathbf{x}} \left\{ \mathbf{c}^\intercal \mathbf{x} + \sum_{k=1}^{K} p_k Q(\mathbf{x}, \boldsymbol{\xi}_k) : \mathbf{x} \in \mathcal{X} \right\}$$

First-stage cost; Expected second-stage cost; First-stage decisions

Second-Stage Problem Second-stage cost

$$Q(\mathbf{x}, \boldsymbol{\xi}) := \min_{\mathbf{y}} \left\{ F(\mathbf{y}, \boldsymbol{\xi}) : \mathbf{y} \in \mathcal{Y}(\mathbf{x}, \boldsymbol{\xi}) \right\}$$

Second-stage decisions

Objective: Determine optimal first-stage decisions that minimize sum of first-stage cost and worst-case scenario..
Challenge: Exact optimization becomes exponentially harder with the number of observed scenarios, *K*. Integer decision further aggravates intractability.

First-Stage Problem

$$\min_{\mathbf{x}} \left\{ \mathbf{c}^\intercal \mathbf{x} + Q(\mathbf{x})) : \mathbf{x} \in \mathcal{X} \right\}$$

First-stage cost; First-stage decisions; Worst-case cost

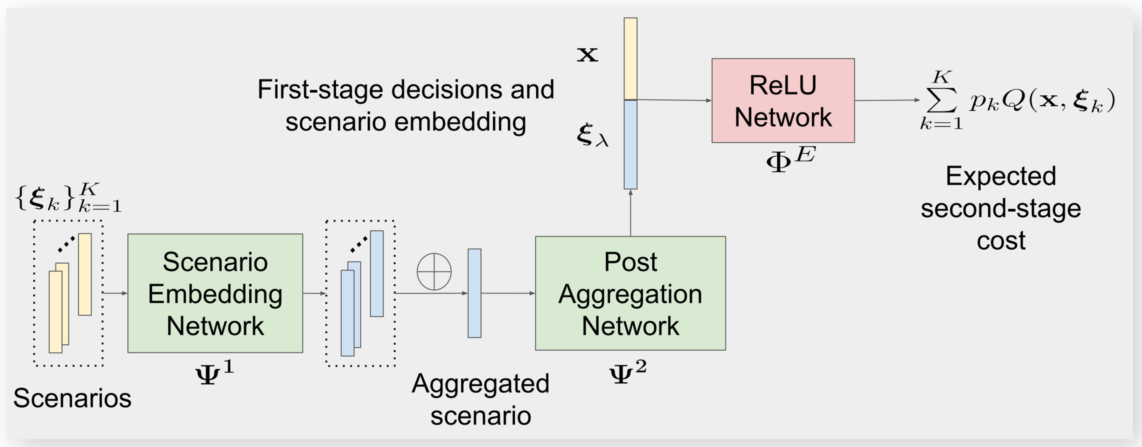
Second-Stage Problem Second-stage cost

$$Q(\mathbf{x}) := \max_{\boldsymbol{\xi} \in \Xi} \min_{\mathbf{y}} \left\{ \mathbf{d}(\boldsymbol{\xi})^\intercal \mathbf{y} : \mathbf{y} \in \mathcal{Y}(\mathbf{x}, \boldsymbol{\xi}) \right\}$$

Second-stage decisions

Methodology

Learning Model

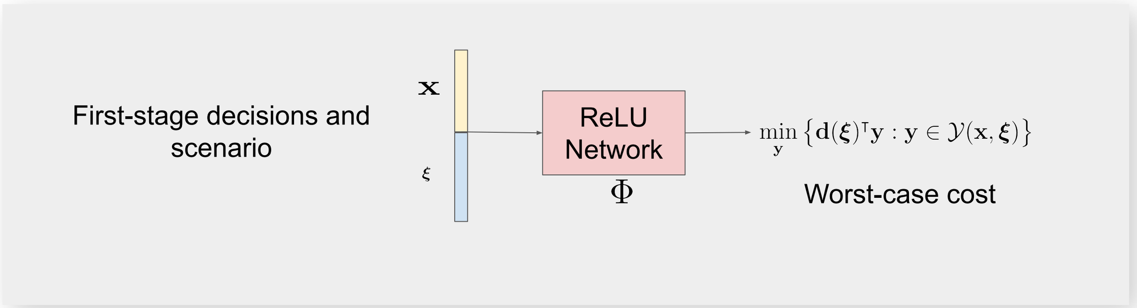


Surrogate Optimization Model

$$\min_{\mathbf{x}} \left\{ \mathbf{c}^\intercal \mathbf{x} + \mathrm{NN}(\mathbf{x}, \{\boldsymbol{\xi}_k\}_{k=1}^{K}) : \mathbf{x} \in \mathcal{X} \right\}$$

Add details about how optimization problem is solved and pros/cons

Learning Model



Surrogate Optimization Model

$$\min_{\mathbf{x}} \left\{ \mathbf{c}^\intercal \mathbf{x} + \eta : \mathbf{x} \in \mathcal{X}, \eta \geq \mathrm{NN}(\mathbf{x}, \boldsymbol{\xi}), \forall \boldsymbol{\xi} \in \Xi \right\}$$

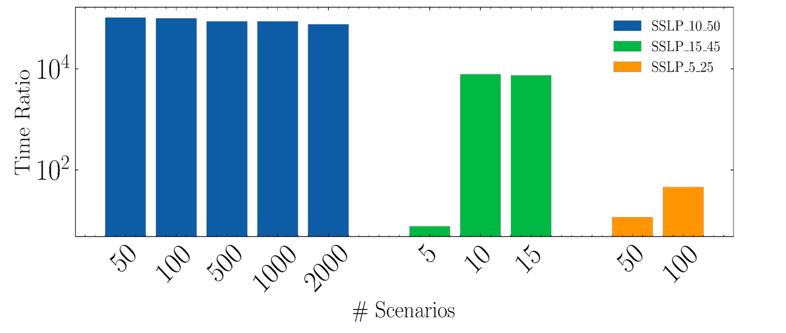
Add details about how optimization problem is solved and pros/cons

Experimental Results

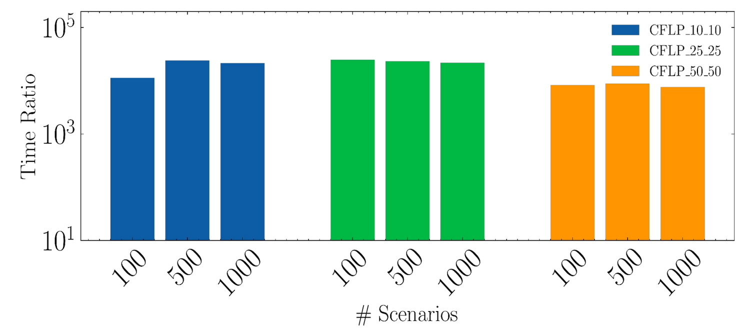
Gap: Mean % difference in solution quality relative to baseline (lower is better).

Bars: Reduction in computing time over baseline (higher is better).

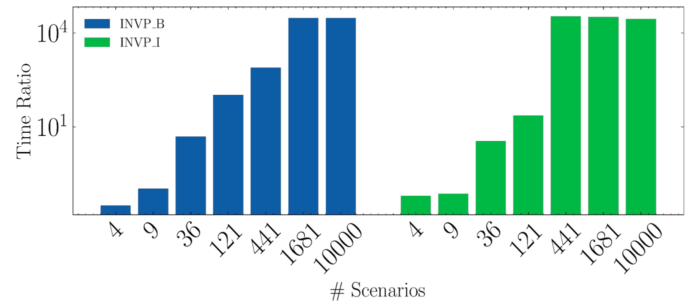
Stochastic Server Location Problem (Gap: -14.91%)



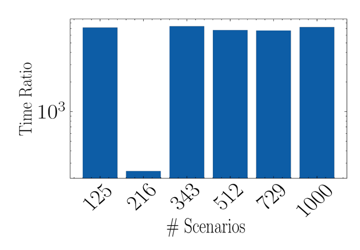
Capacitated Facility Location Problem (Gap: -2.93%)



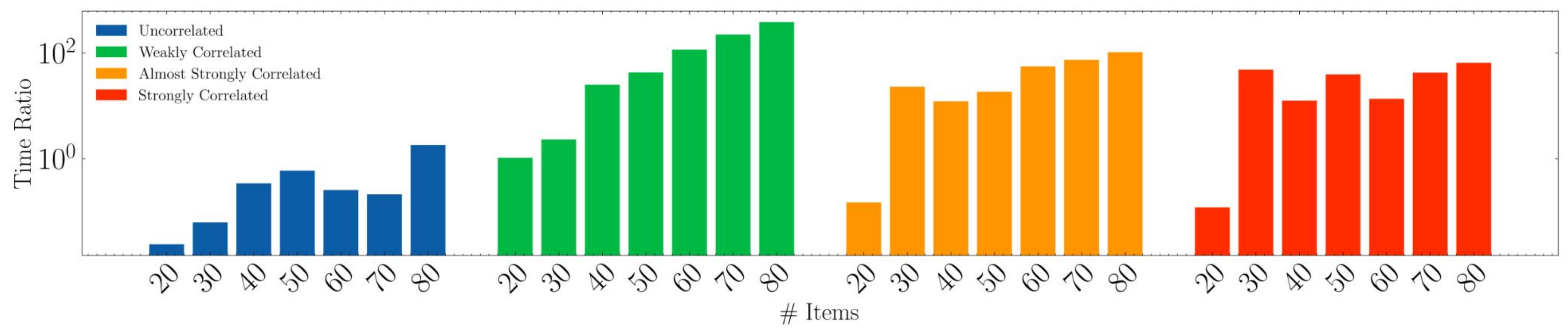
Investment Problem (Gap: 3.82%)



Pooling Problem (Gap: 4.82%)



Add expereimentsal results



Average Optimality Gaps:
- Uncorrelated: 3.96%
- Weakly Correlated: 7.25%
- Almost Strongly Correlated: 5.63%
- Strongly Correlated: 4.69%

ations

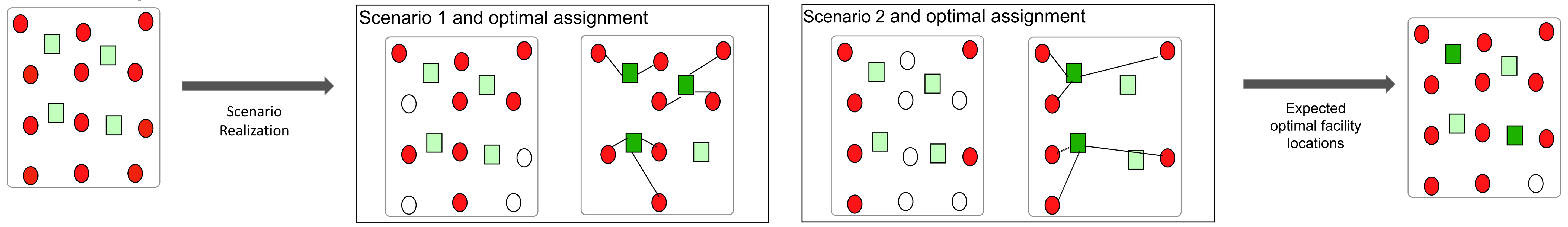
A Unified Machine Learning Framework for Optimization under Uncertainty

Justin Dumouchelle
Mechanical & Industrial Engineering
UNIVERSITY OF TORONTO

Learning for **Two-Stage Stochastic Programming (2SP)**

Stochastic Facility Location Problem

Objective: Determine the optimal set of facilities to construct given uncertainty in client requests.



Formulation

Objective: Determine optimal first-stage decisions that minimize sum of first-stage cost and expected second-stage cost.
Challenge: Exact optimization becomes exponentially harder with the number of observed scenarios, *K*. Integer decision further aggravates intractability.

First-Stage Problem

$$\min_{\mathbf{x}} \Big\{ \underbrace{\mathbf{c}^\intercal \mathbf{x}}_{\text{First-stage cost}} + \underbrace{\sum_{k=1}^{K} p_k Q(\mathbf{x}, \boldsymbol{\xi}_k)}_{\text{Expected second-stage cost}} : \mathbf{x} \in \mathcal{X} \Big\}$$

First-stage decisions

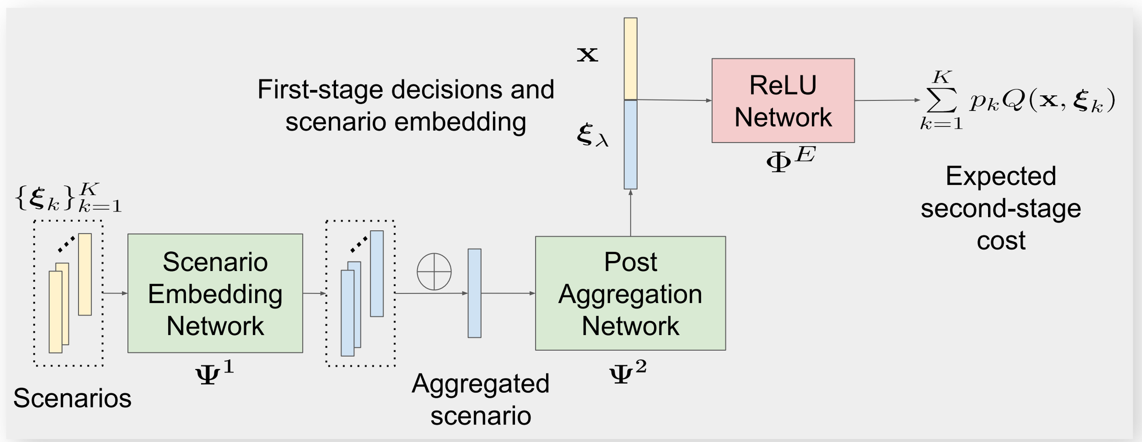
Second-Stage Problem

$$Q(\mathbf{x}, \boldsymbol{\xi}) := \min_{\mathbf{y}} \{ \underbrace{F(\mathbf{y}, \boldsymbol{\xi})}_{\text{Second-stage cost}} : \mathbf{y} \in \mathcal{Y}(\mathbf{x}, \boldsymbol{\xi}) \}$$

Second-stage decisions

Methodology

Learning Model



Surrogate Optimization Model

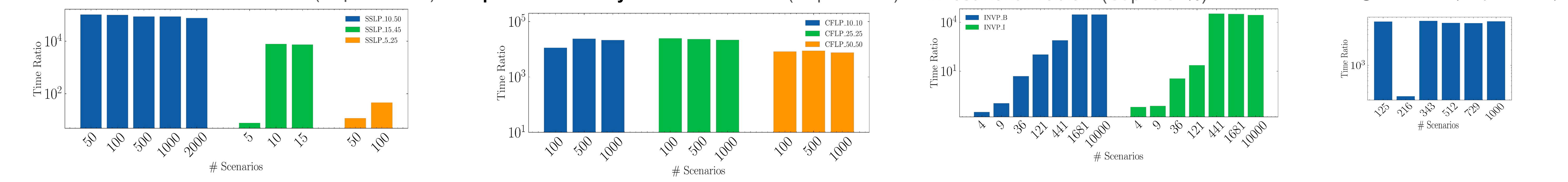
$$\min_{\mathbf{x}} \{ \mathbf{c}^\intercal \mathbf{x} + \mathrm{NN}(\mathbf{x}, \{\boldsymbol{\xi}_k\}_{k=1}^{K}) : \mathbf{x} \in \mathcal{X} \}$$

Add details about how optimization problem is solved and pros/cons

Experimental Results

Gap: Mean % difference in solution quality relative to baseline (lower is better). **Bars**: Reduction in computing time over baseline (higher is better).

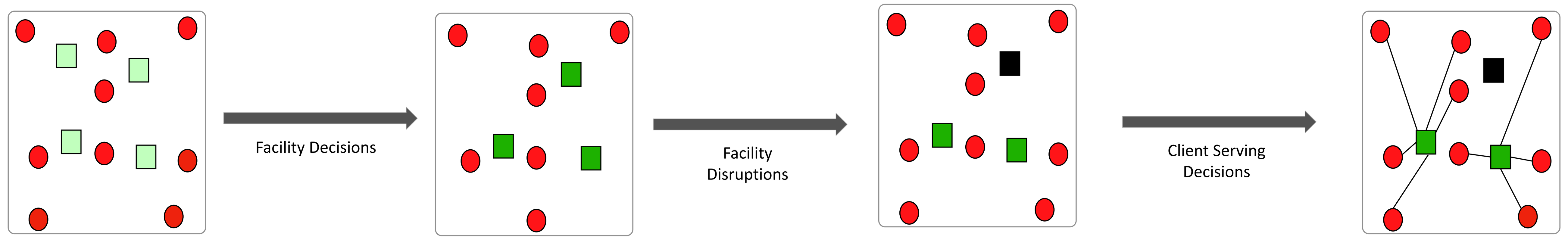
Stochastic Server Location Problem (Gap: -14.91%) **Capacitated Facility Location Problem** (Gap: -2.93%) **Investment Problem** (Gap: 3.82%) **Pooling Problem** (Gap: 4.82%)



Learning for **Adjustable Robust Optimization (ARO)**

Robust Facility Location Problem

Objective: Determine the optimal set of facilities to open under the worst-case facility disruption.



Formulation

Objective: Determine optimal first-stage decisions that minimize sum of first-stage cost and worst-case scenario..
Challenge: Exact optimization becomes exponentially harder with the number of observed scenarios, *K*. Integer decision further aggravates intractability.

First-Stage Problem

$$\min_{\mathbf{x}} \{ \underbrace{\mathbf{c}^\intercal \mathbf{x}}_{\text{First-stage cost}} + \underbrace{Q(\mathbf{x})}_{\text{Worst-case cost}}) : \mathbf{x} \in \mathcal{X} \}$$

First-stage decisions

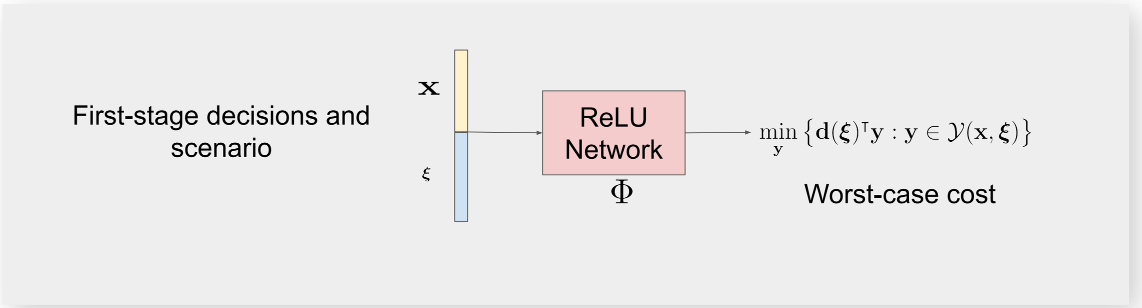
Second-Stage Problem

$$Q(\mathbf{x}) := \max_{\boldsymbol{\xi} \in \Xi} \min_{\mathbf{y}} \{ \underbrace{\mathbf{d}(\boldsymbol{\xi})^\intercal \mathbf{y}}_{\text{Second-stage cost}} : \mathbf{y} \in \mathcal{Y}(\mathbf{x}, \boldsymbol{\xi}) \}$$

Second-stage decisions

Methodology

Learning Model



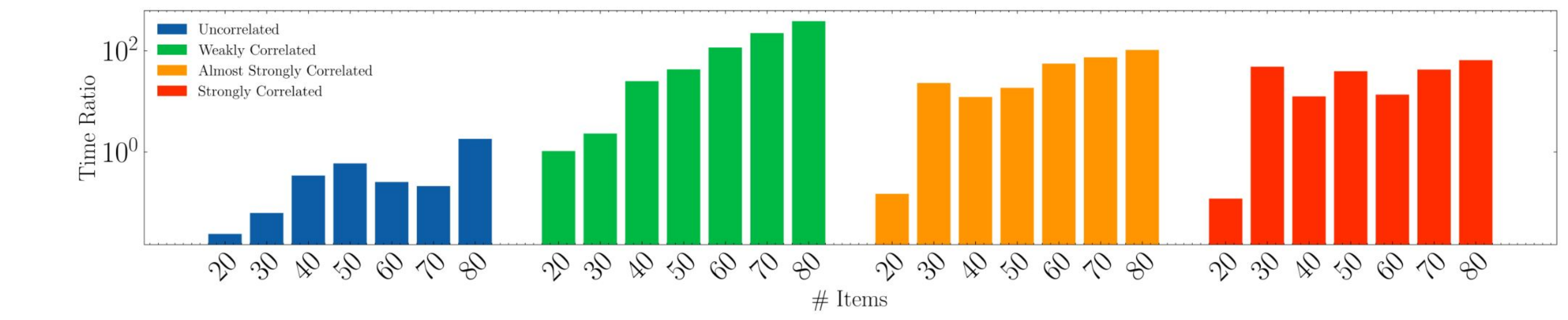
Surrogate Optimization Model

$$\min_{\mathbf{x}} \{ \mathbf{c}^\intercal \mathbf{x} + \eta : \mathbf{x} \in \mathcal{X}, \eta \geq \mathrm{NN}(\mathbf{x}, \boldsymbol{\xi}), \forall \boldsymbol{\xi} \in \Xi \}$$

Add details about how optimization problem is solved and pros/cons

Experimental Results

Add expereimentsal results



Average Optimality Gaps:
- Uncorrelated: 3.96%
- Weakly Correlated: 7.25%
- Almost Strongly Correlated: 5.63%
- Strongly Correlated: 4.69%

A Unified Machine Learning Framework for Optimization Under Uncertainty

Justin Dumouchelle

Learning for **Two-Stage Stochastic Programming (2SP)**

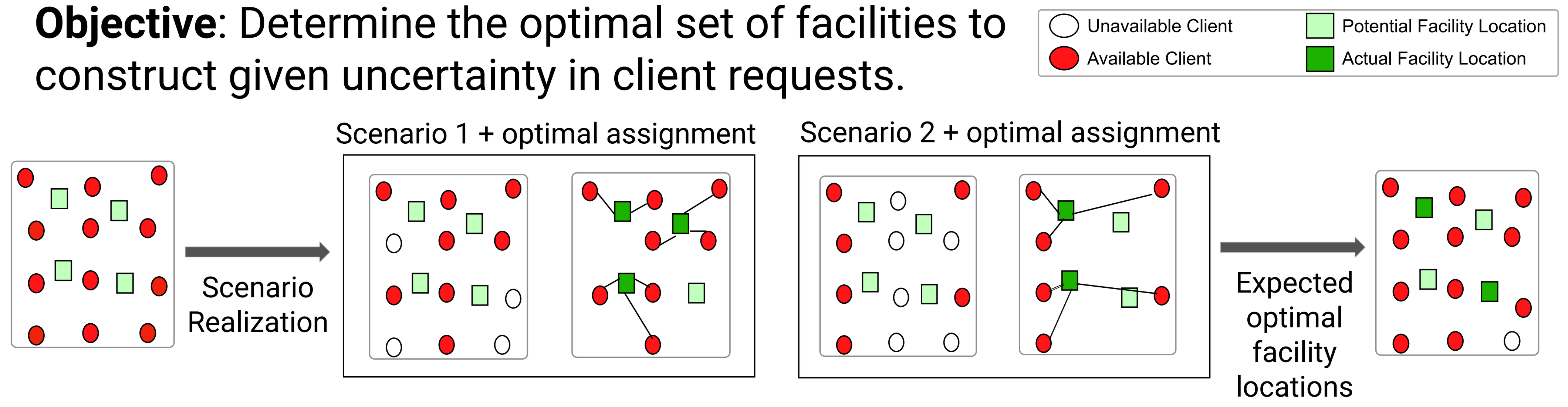
Published at NeurIPS 2022. Joint work Elias Khalil, Merve Bodur, and Rahul Patel

Learning for **Adjustable Robust Optimization (ARO)**

Under review (you can updated this to an arxiv link for the conference)

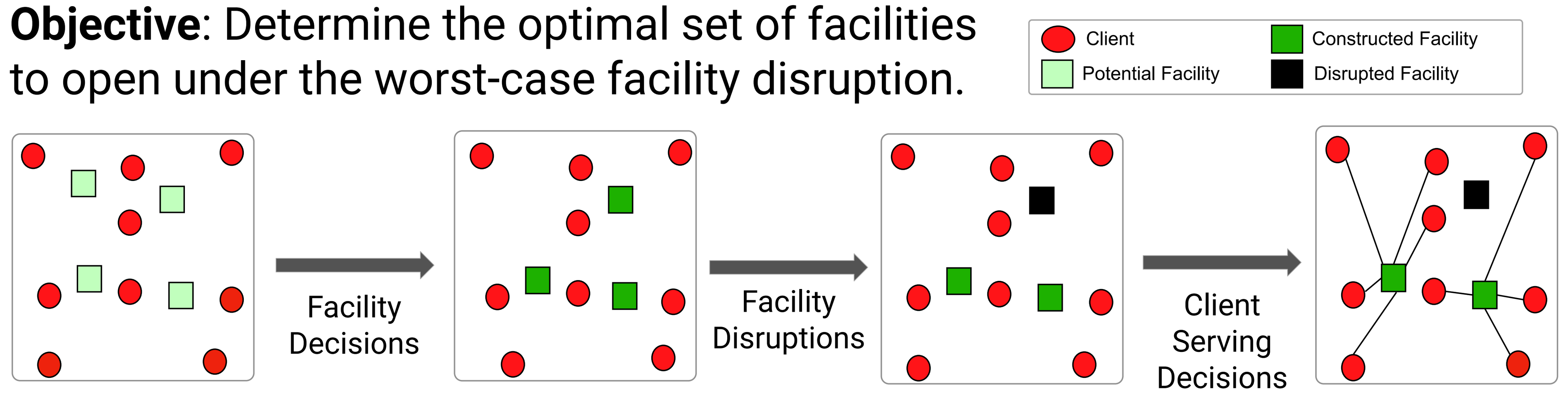
Stochastic Facility Location Problem

Objective: Determine the optimal set of facilities to construct given uncertainty in client requests.



Robust Facility Location Problem

Objective: Determine the optimal set of facilities to open under the worst-case facility disruption.



Overall framework

Data Collection:
- Sample decisions + uncertainty
- Solve subproblems with off-the-shelf solvers

Supervised Learning:
- Train an NN using off-the-shelf ML packages

Evaluation on new instances:
- Solve the surrogate MILP with an off-the-shelf solver

Optimization Formulation

Objective: Determine optimal first-stage decisions that minimize sum of first-stage cost and expected second-stage cost.

Challenge: Exact optimization becomes exponentially harder with the number of observed scenarios, *K*. Integer decision further aggravates intractability.

First-Stage Problem

$$\min_{\mathbf{x}} \left\{ \underbrace{\boxed{\mathbf{c}^\intercal \mathbf{x}}}_{\text{First-stage cost}} + \underbrace{\boxed{\sum_{k=1}^{K} p_k Q(\mathbf{x}, \xi_k)}}_{\text{Expected second-stage cost}} : \mathbf{x} \in \mathcal{X} \right\}$$

First-stage decisions: $\mathbf{x}$

Second-Stage Problem

$$Q(\mathbf{x}, \xi) := \min_{\mathbf{y}} \left\{ \underbrace{\boxed{F(\mathbf{y}, \xi)}}_{\text{Second-stage cost}} : \mathbf{y} \in \mathcal{Y}(\mathbf{x}, \xi) \right\}$$

Second-stage decisions: $\mathbf{y}$

Objective: Determine optimal first-stage decisions that minimize first-stage cost and under the worst-case cost.

Challenge:

First-Stage Problem

$$\min_{\mathbf{x}} \left\{ \underbrace{\boxed{\mathbf{c}^\intercal \mathbf{x}}}_{\text{First-stage cost}} + \underbrace{\boxed{Q(\mathbf{x})}}_{\text{Worst-case cost}} \right\} : \mathbf{x} \in \mathcal{X} \right\}$$

First-stage decisions: $\mathbf{x}$

Second-Stage Problem

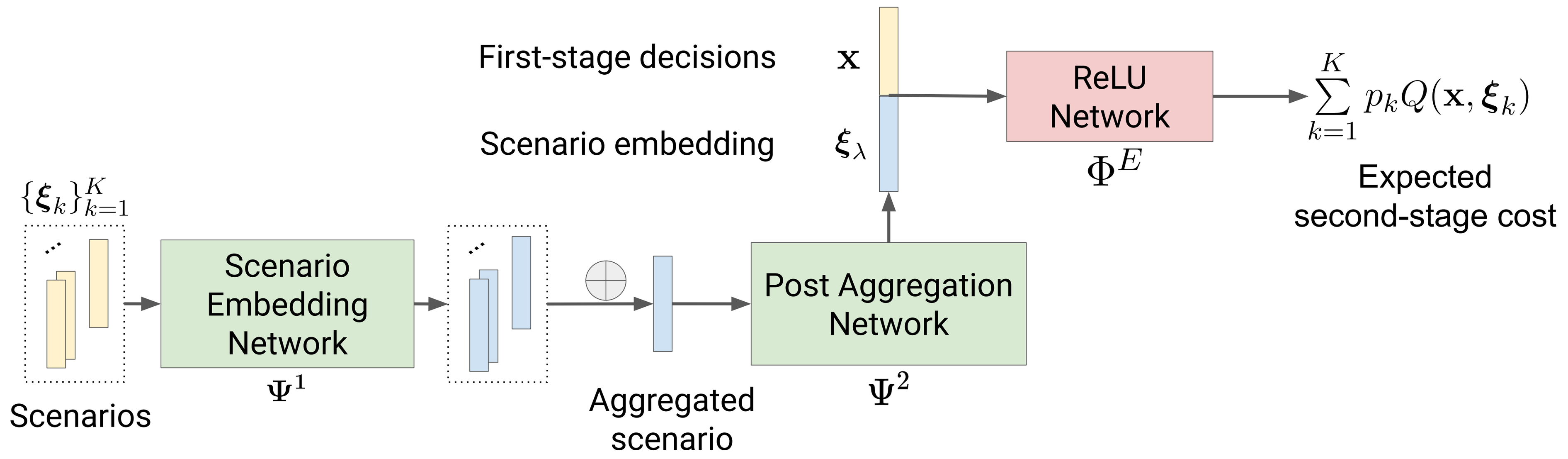
$$Q(\mathbf{x}) := \max_{\boxed{\xi \in \Xi}} \min_{\boxed{\mathbf{y}}} \left\{ \underbrace{\boxed{\mathbf{d}(\xi)^\intercal \mathbf{y}}}_{\text{Second-stage cost}} : \mathbf{y} \in \mathcal{Y}(\mathbf{x}, \xi) \right\}$$

Worst-case uncertainty: $\xi \in \Xi$; Second-stage decisions: $\mathbf{y}$

Machine Learning Methodology

ML Solution: Replace the expected stage-cost value function with a neural network approximation.

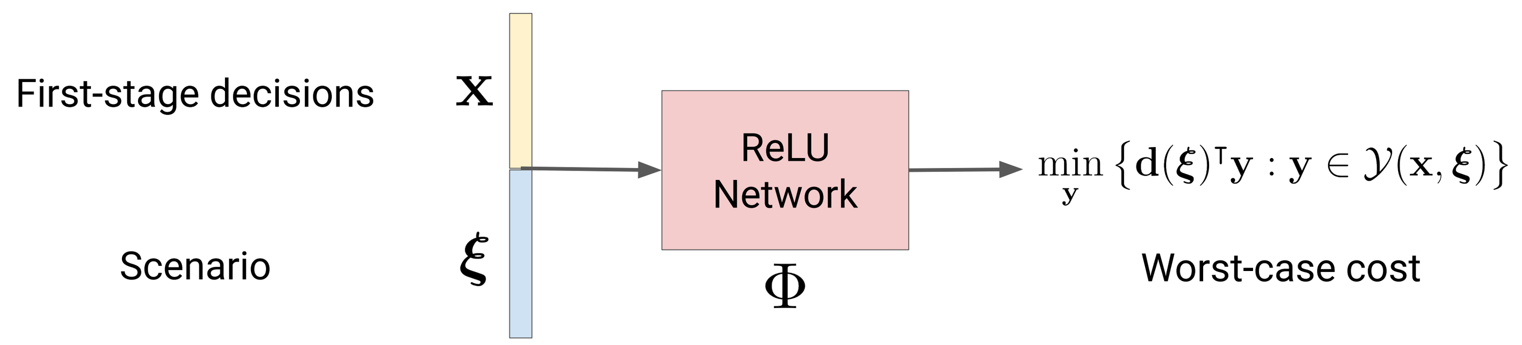
Neural Network Architecture:



Surrogate Optimization Model: $\min_{\mathbf{x}} \left\{ \mathbf{c}^\intercal \mathbf{x} + \mathrm{NN}(\mathbf{x}, \{\xi_k\}_{k=1}^K) : \mathbf{x} \in \mathcal{X} \right\}$

ML Solution: Replace the worst-case cost with a neural network approximation.

Neural Network Architecture:

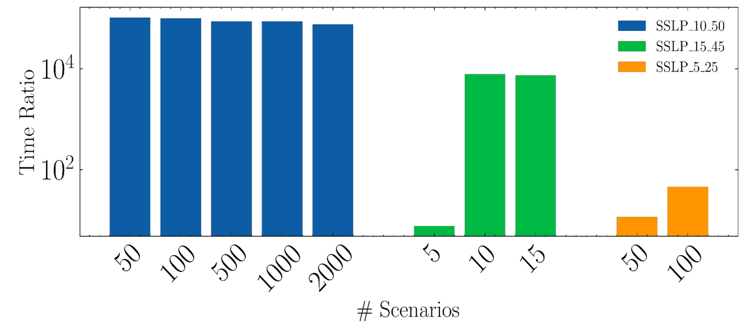


Surrogate Optimization Model: $\min_{\mathbf{x}} \left\{ \mathbf{c}^\intercal \mathbf{x} + \eta : \mathbf{x} \in \mathcal{X}, \eta \geq \mathrm{NN}(\mathbf{x}, \xi), \forall \xi \in \Xi \right\}$

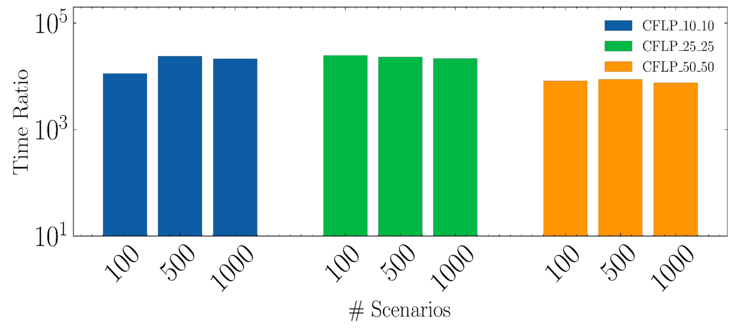
Experimental Results

Stochastic Programming

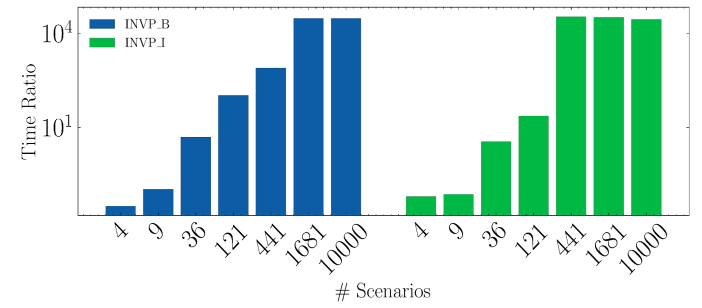
Stochastic Server Location Problem (Gap: -14.91%)



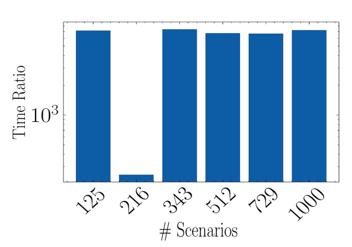
Capacitated Facility Location Problem (Gap: -2.93%)



Investment Problem (Gap: 3.82%)

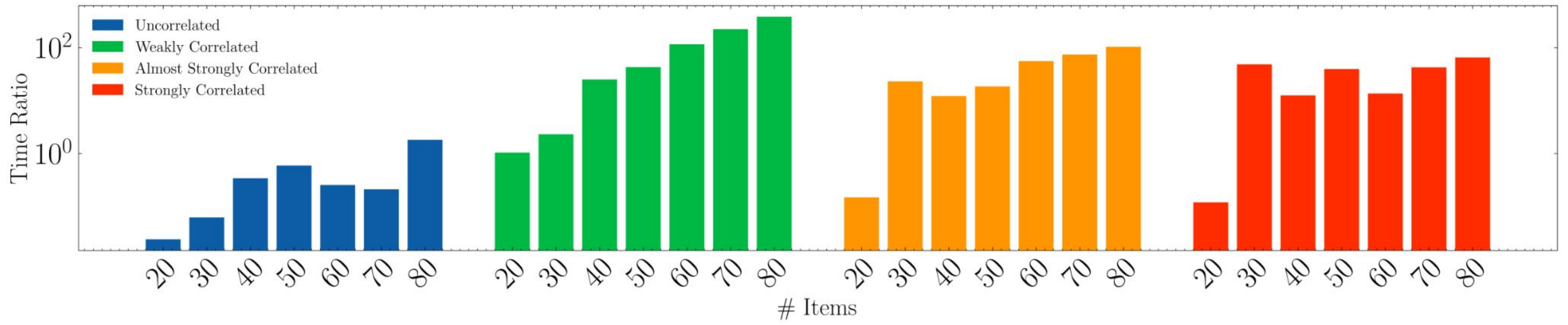


Pooling Problem (Gap: 4.82%)



Reduction factor in computing time over baseline (WHICH) (higher is better).

Robust Optimization



Average Optimality Gaps:
- Uncorrelated: 3.96%
- Weakly Correlated: 7.25%
- Almost Strongly Correlated: 5.63%
- Strongly Correlated: 4.69%

Mechanical & Industrial Engineering
UNIVERSITY OF TORONTO