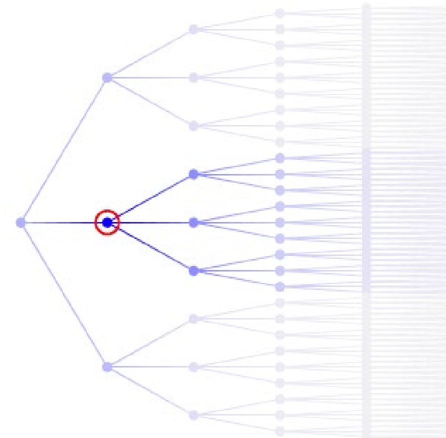
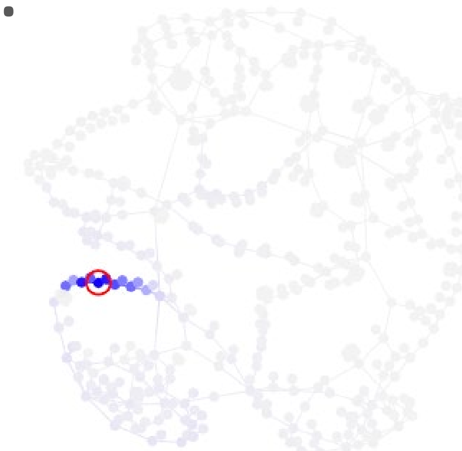


A Graph-Theoretic Perspective on Optimization:

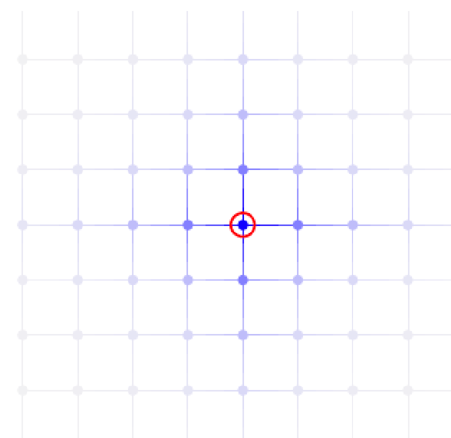
Theory, Modeling, Algorithms, and Applications

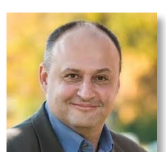


Victor M. Zavala

Department of Chemical & Biological Engineering
University of Wisconsin-Madison

Mathematics and Computer Science Division
Argonne National Laboratory



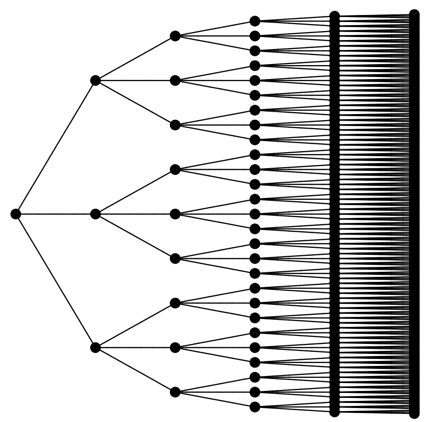
Work with: Sungho Shin, Mihai Anitescu, Jordan Jalving, David Cole

Motivation

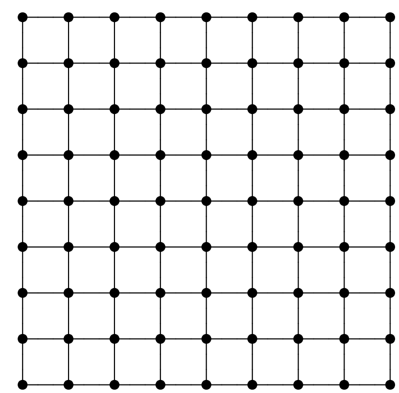
What is a General Math Abstraction of these Problem Classes?

Dynamic Optimization

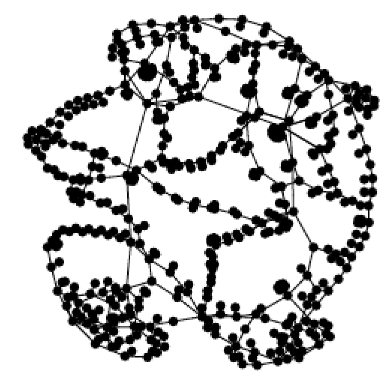
Stochastic Optimization



PDE Optimization

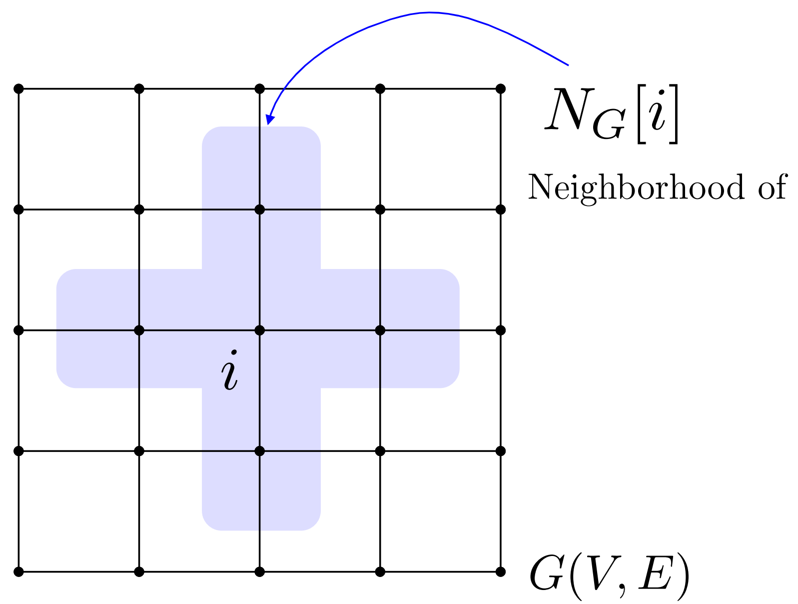


Network Optimization



Graph-Structured Nonlinear Programs (gsNLP)

$$\begin{aligned}\min_{\{z_i\}_{i\in V}} \quad & \sum_{i\in V} f_i(\{z_j\}_{j\in N_G[i]}, p_i) \\ \text{s.t.} \quad & g_i(\{z_j\}_{j\in N_G[i]}, p_i) \geq 0, \; i \in V\end{aligned}$$



- Problem domain given by graph $G(V, E)$.

- Each node $i \in V$ has local objective, constraints, and data.

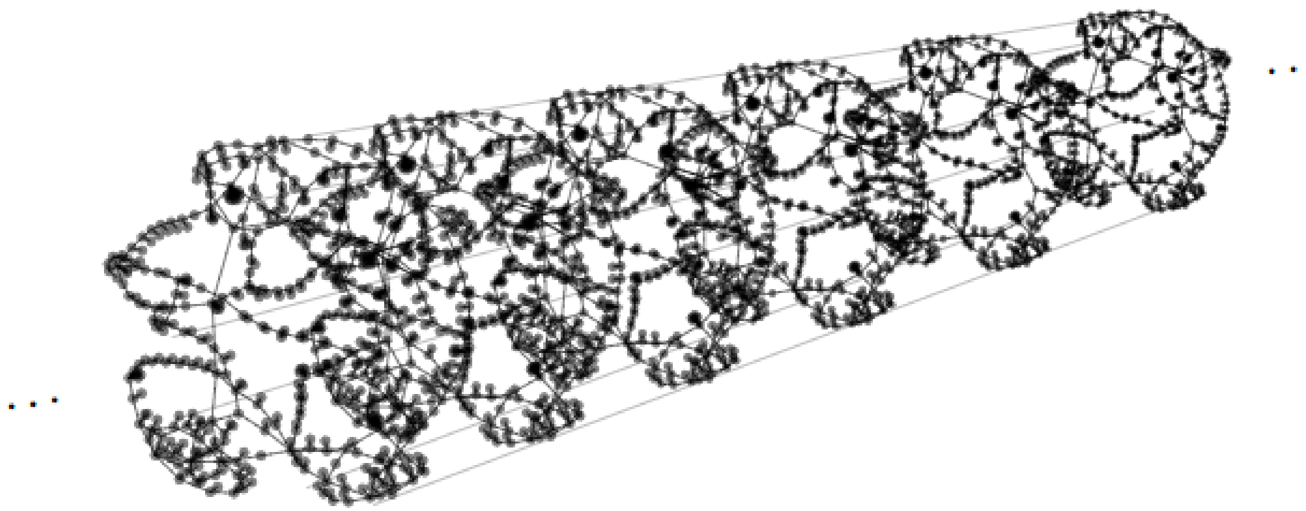
- Each node $i \in V$ has local variables (e.g., states and controls).

- Node $i \in V$ is coupled to its neighbors $N_G[i] \subseteq V$



Dynamic Network Optimization

(e.g., natural gas, power, water, transportation networks)



Exponential Decay of Sensitivity (EDS)

Sensitivity of NLPs (Classical)

How does solution z^* change when data p changes?

$$p, p' \Longrightarrow \boxed{\begin{aligned} \min_{z} \quad & f(z,p) \\ \text{s.t.} \quad & g(z,p) \geq 0 \end{aligned}} \Longrightarrow z^*(p), z^*(p')$$

Assumptions:

- Consider base data p and perturbation p'.
- Assume base solution $z^*(p)$ satisfies SSOC and LICQ (regularity conditions).

Theorem: There exists a continuous solution mapping $z^*(\cdot)$ and:

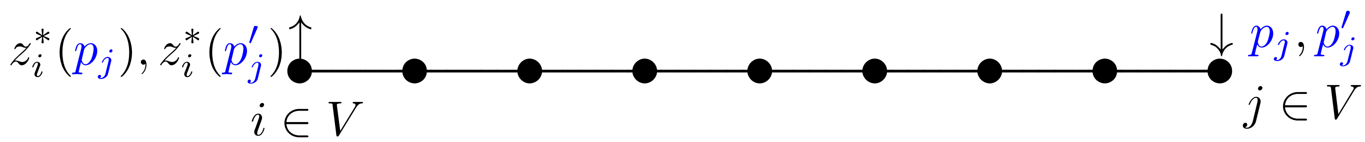
$$\|z^*(p) - z^*(p')\| \leq C\|p - p'\|$$

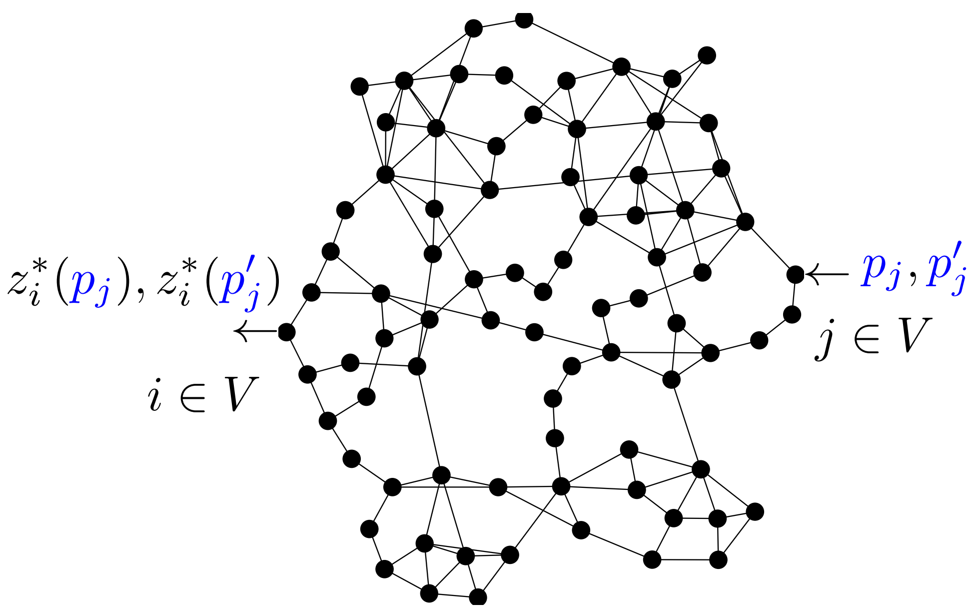
with Lipschitz Constant C.

Biegler, L. T. (2013). A survey on sensitivity-based nonlinear model predictive control. *IFAC Proceedings Volumes*, *46*(32), 499-510.
Robinson, S. M. (1980). Strongly regular generalized equations. Mathematics of Operations Research, 5(1), 43-62.

Sensitivity of gsNLPs

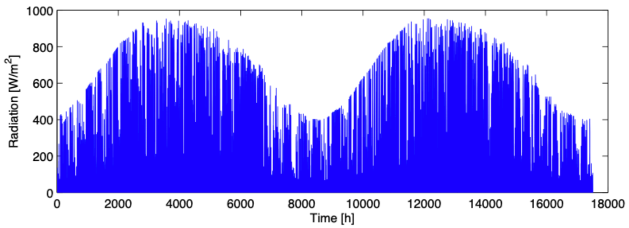
How does solution z_i^* (at node $i \in V$) change when data p_j (at node $j \in V$) changes?

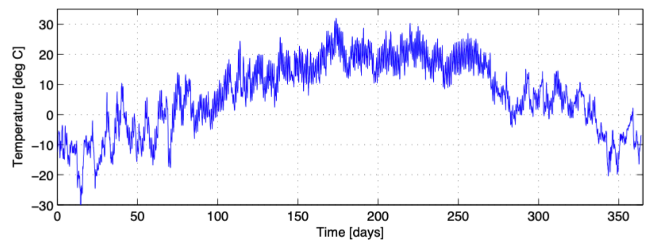


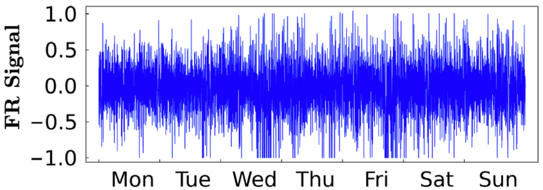


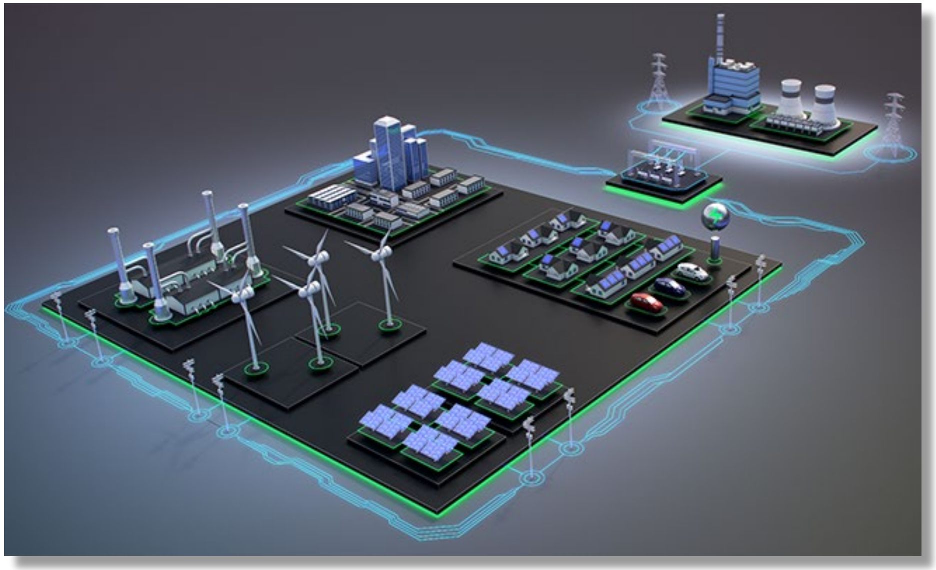
Na, S., & Anitescu, M. (2020). Exponential decay in the sensitivity analysis of nonlinear dynamic programming. *SIAM Journal on Optimization*, *30*(2), 1527-1554.
Shin, S., Anitescu, M., & Zavala, V. M. (2021). Exponential Decay of Sensitivity in Graph-Structured Nonlinear Programs. arXiv preprint arXiv:2101.03067.

Exponential Decay of Sensitivity (EDS): Temporal



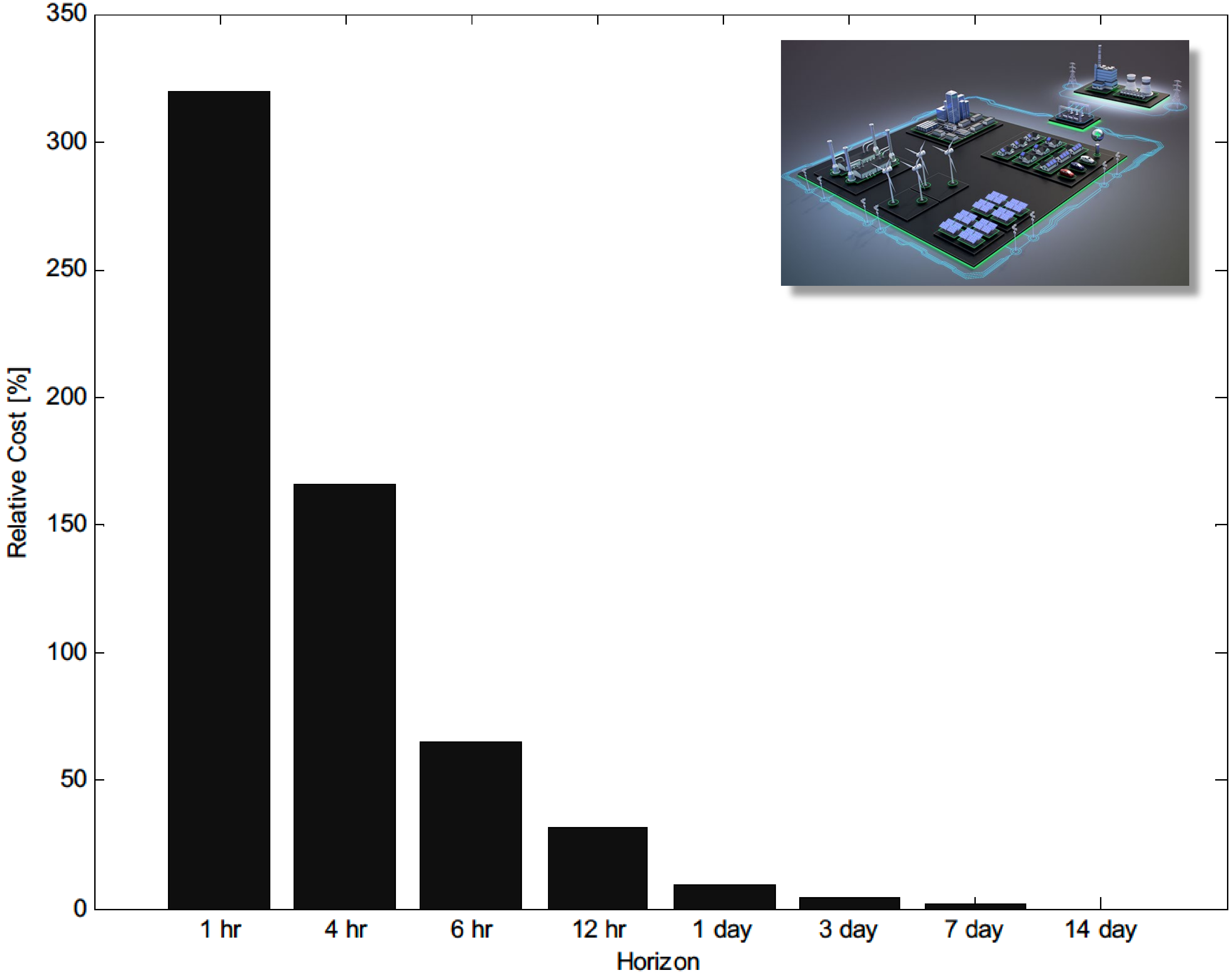






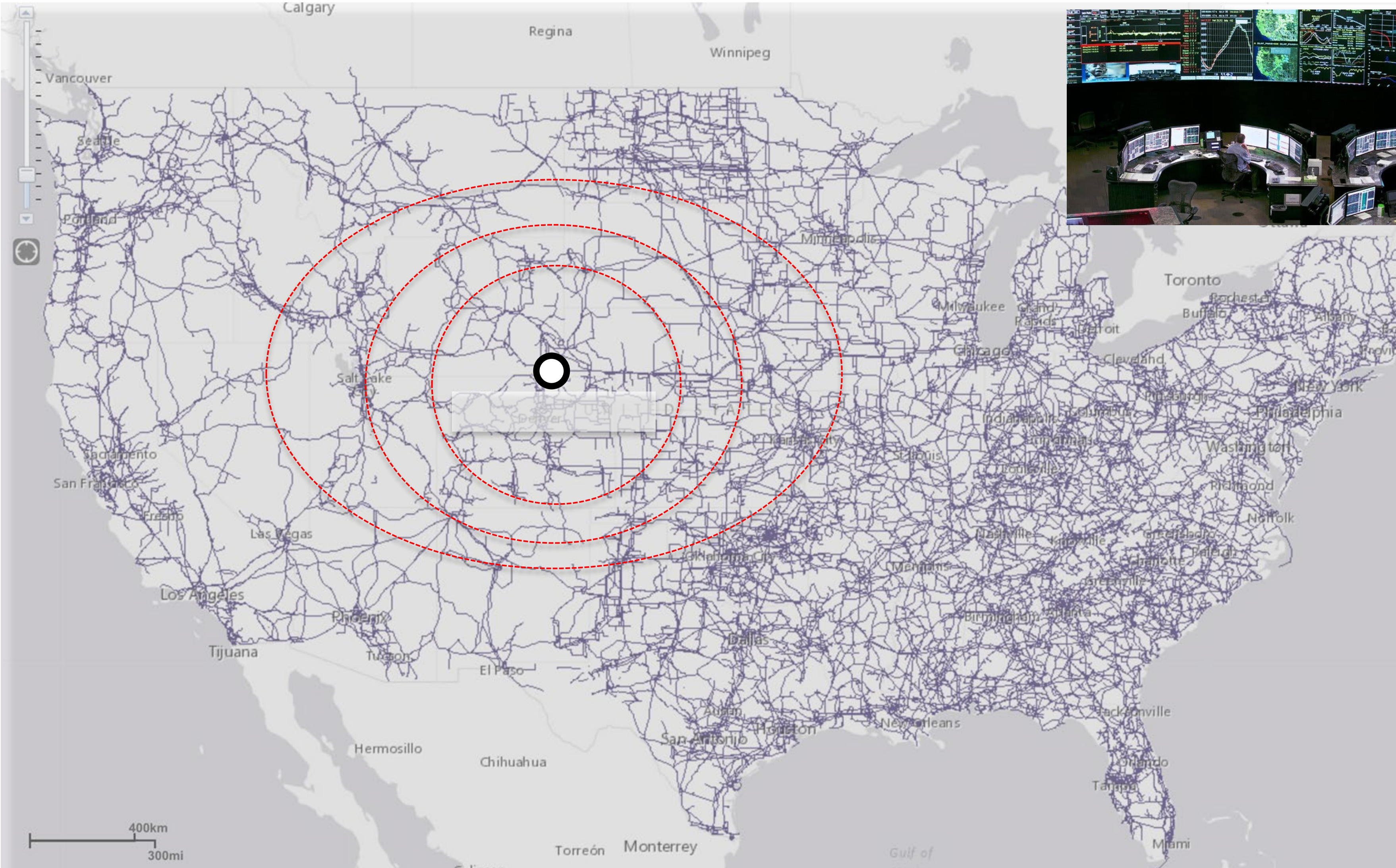
$$
\begin{aligned}
\min_{u(\cdot)} \quad & \int_0^T \varphi(z(\tau), u(\tau), p(\tau))d\tau \\
\text{s.t.} \quad & \dot{z}(\tau) = f(z(\tau), u(\tau), p(\tau))
\end{aligned}
$$

Exponential Decay of Sensitivity (EDS): Temporal



Zavala, V. M., Anitescu, M., & Krause, T. (2009). On the optimal on-line management of photovoltaic-hydrogen hybrid energy systems. In *CACE* (Vol. 27, pp. 1953-1958)

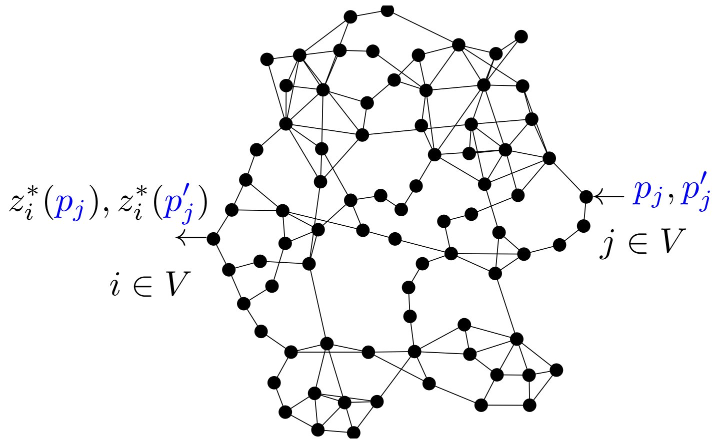
Exponential Decay of Sensitivity (EDS): Spatial



EDS in gsNLPs

How does solution z_i^* (at node $i \in V$) change
when data p_j (at node $j \in V$) changes?

$$\min_{\{z_i\}_{i\in V}} \sum_{i\in V} f_i(\{z_j\}_{j\in N_G[i]}, p_i)$$
$$\text{s.t.} \quad g_i(\{z_j\}_{j\in N_G[i]}, p_i) \geq 0, \ i \in V$$



Assumptions:

- Assume base solution $z^*(p)$ satisfies SSOC and LICQ (regularity conditions).

Theorem: The solutions at node $i \in V$ satisfy:

$$\|z_i^*(p_j) - z_i^*(p_j')\| \leq C\rho^{d(i,j)}\|p_j - p_j'\|$$

with $\rho \in (0,1)$, C independent of size of G, and $d(i,j)$ is geodesic distance.

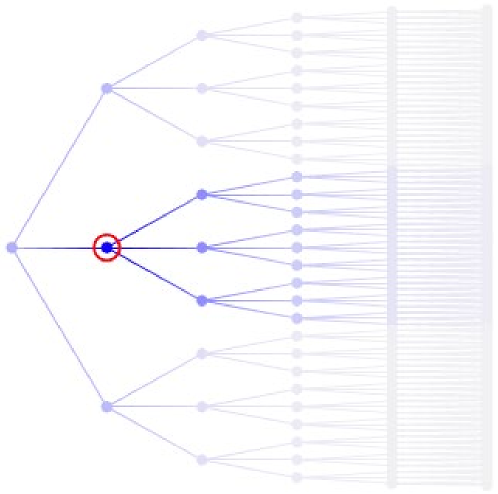
Key: - Solution Sensitivity Decays *Exponentially* with Geodesic Distance.

- Result Obtained from Sparsity Analysis of the KKT Matrix Inverse

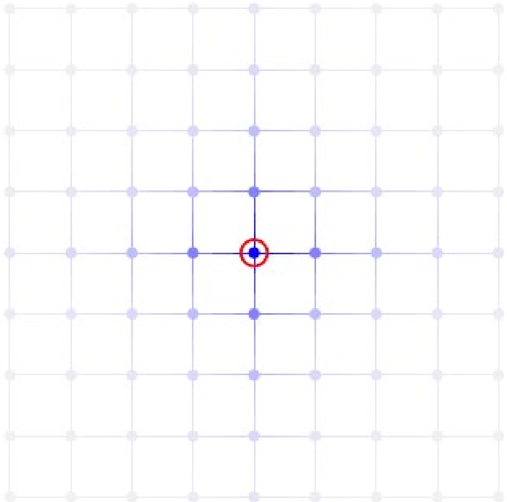
EDS in gsNLPs

Dynamic Optimization

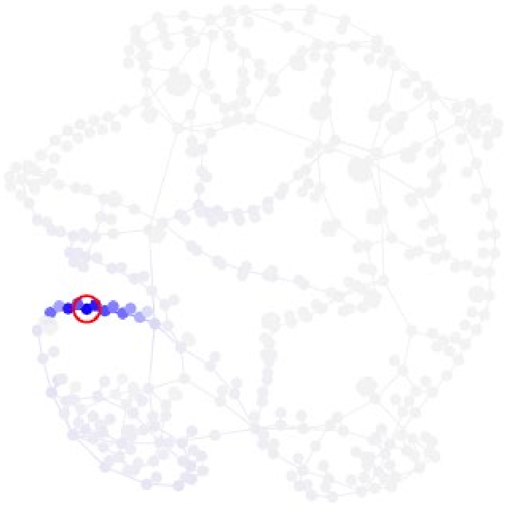
Stochastic Optimization



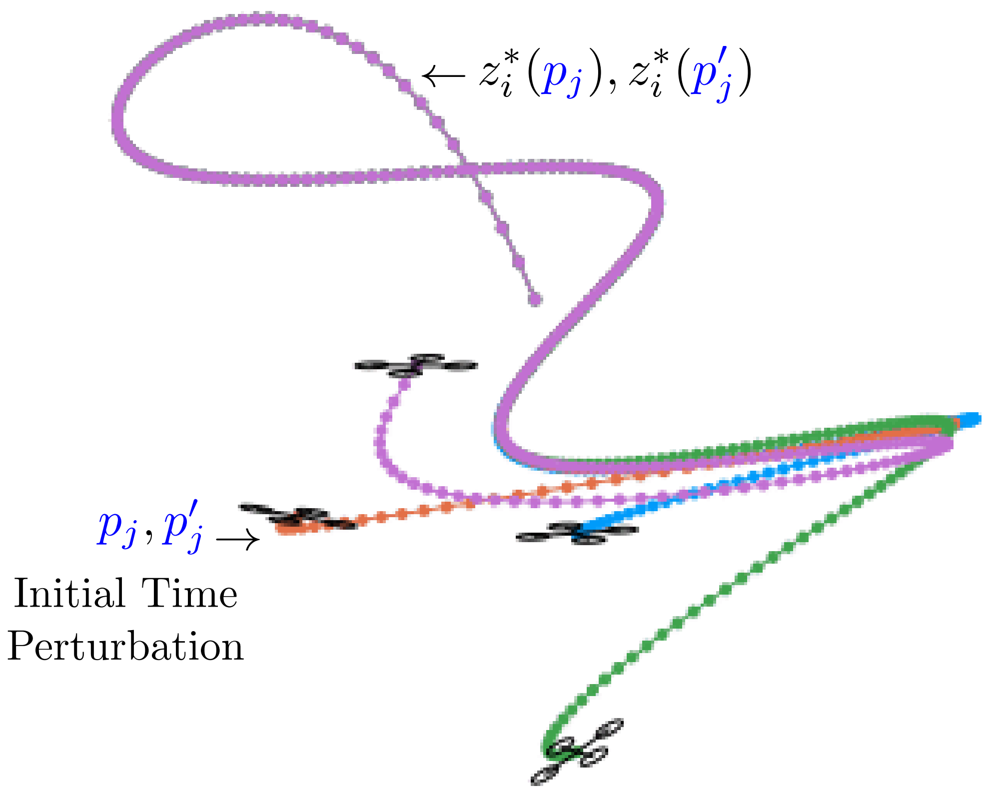
PDE Optimization



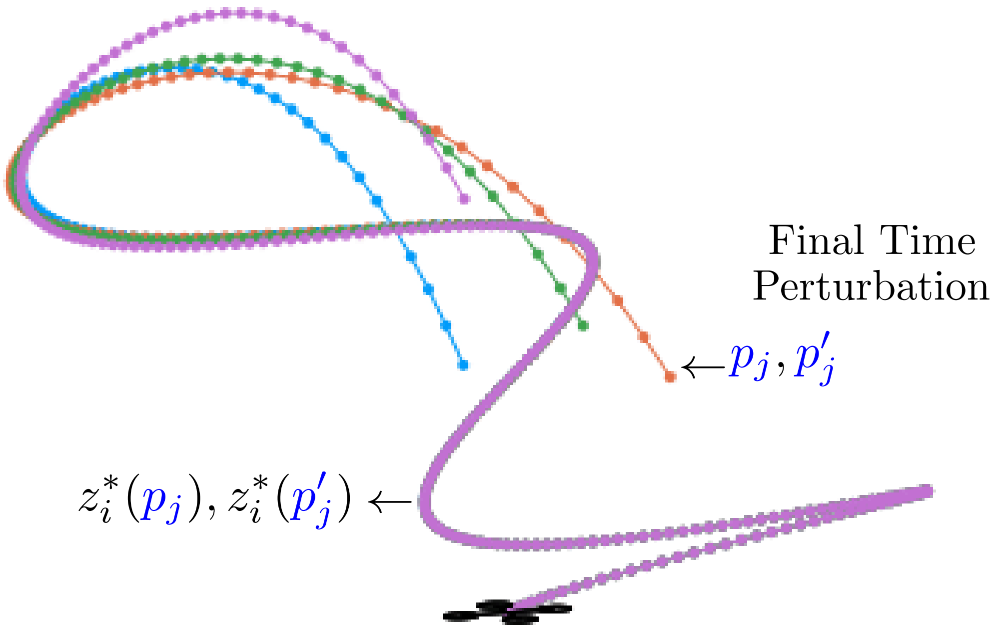
Network Optimization



EDS in gsNLPs



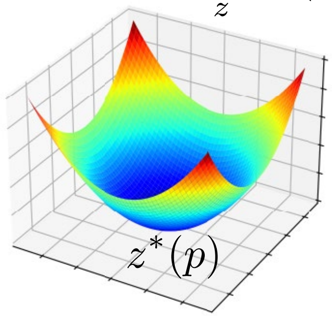
EDS in gsNLPs



EDS in gsNLPs

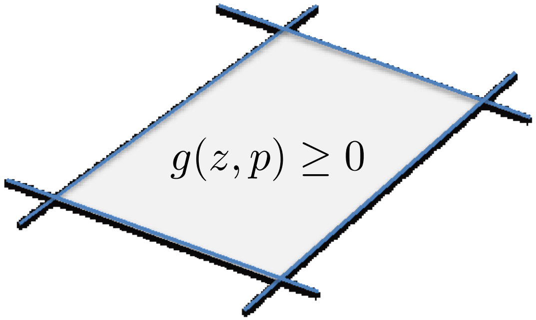
Key: SSOC and LICQ provide Sufficient Conditions for EDS

- Strong Second Order Condition (**SSOC**) Ensures z Impacts Objective
- Smallest Singular Value of Reduced Hessian is a Measure of "Attraction"
- Related to Observability in Optimal Control



$$\sigma_{min}(N^T \nabla^2 \mathcal{L}(z^*(p),p)N) > 0 \quad \text{with} \quad \nabla g(z^*(p),p)\, N = 0$$

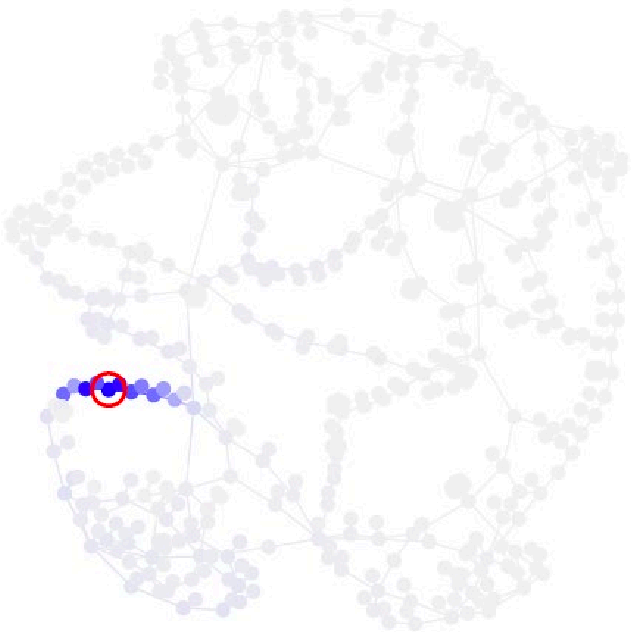
- Linear Independence Constraint Qualification (**LICQ**) Ensures z Impacts Constraints
- Smallest Singular Value of Jacobian is a Measure of "Flexibility"
- Related to Controllability in Optimal Control

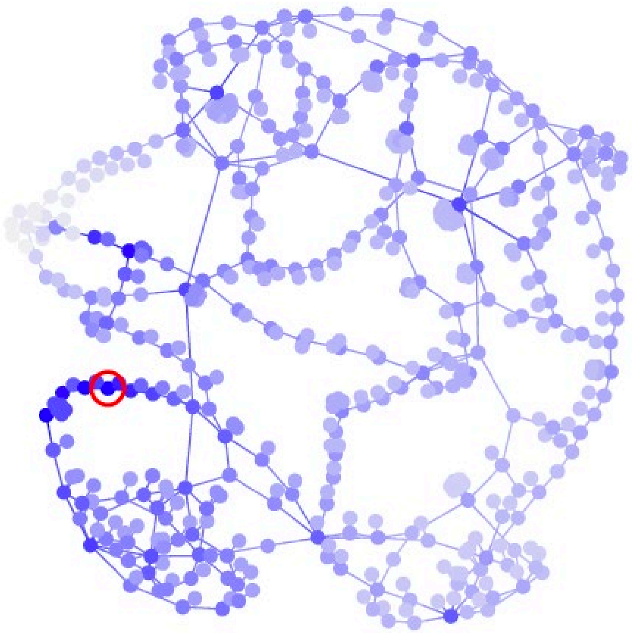


$$\sigma_{min}(\nabla g(z^*(p),p)) > 0$$

Shin, S., & Zavala, V. M. (2021). Controllability and observability imply exponential decay of sensitivity in dynamic optimization. *IFAC-PapersOnLine*, *54*(6), 179-184.

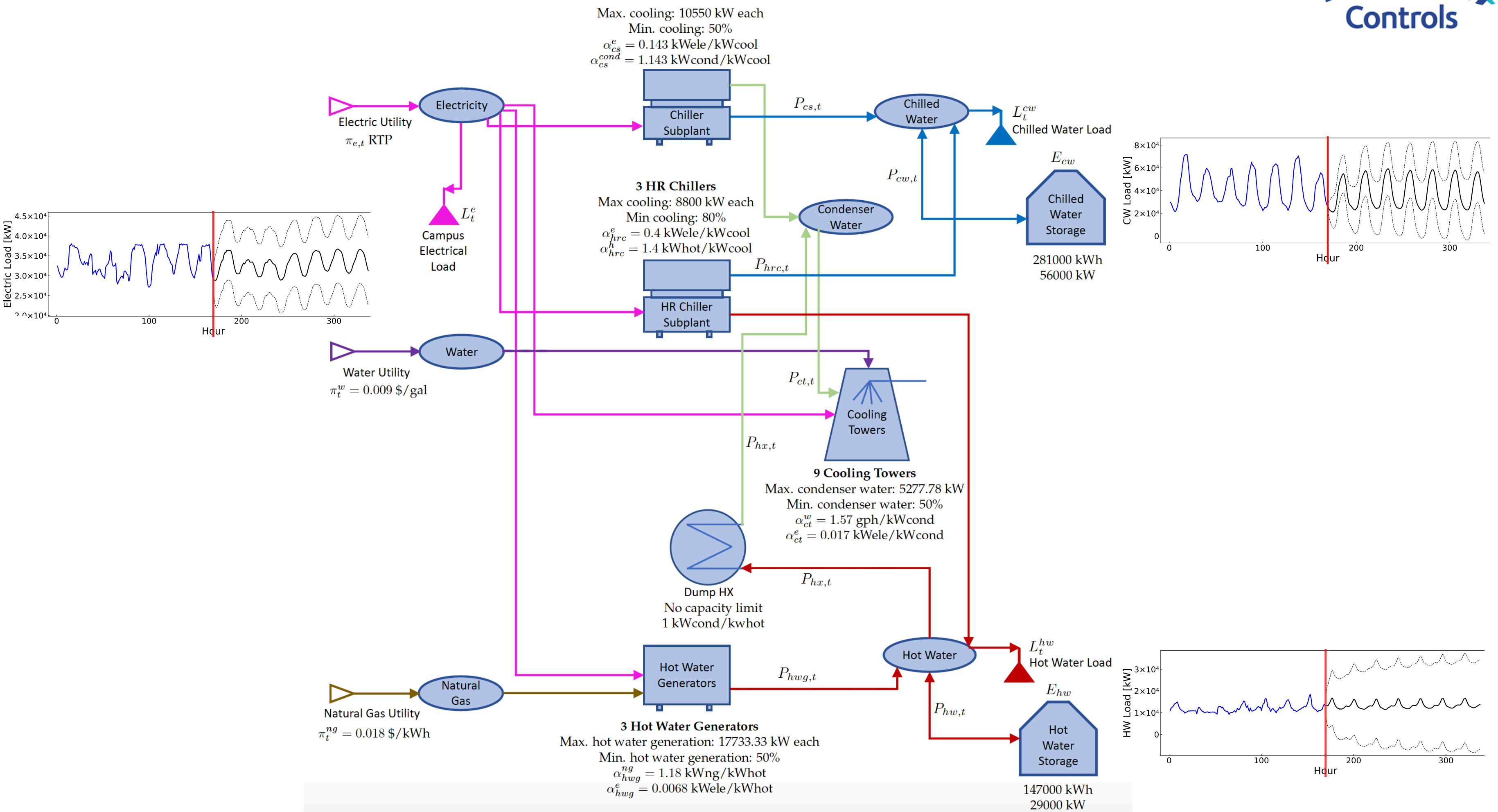
Effect of Regularity





Algorithms

MPC for Utilities Plant



MPC Problem

Objective: Minimize Expected Total Cost and Demand Charge

$$\min \quad \mathbb{E}\left[\sum_{k\in\mathcal{T}}(\pi_k(\xi)^T u_k(\xi) + \mu^T s_k(\xi)) + \frac{\pi^D}{\sigma_t} D(\xi)\right]$$

Load & Energy Balance Constraints:

$$C u_k(\xi) + s_k(\xi) = L_k(\xi)$$

Storage Dynamics:

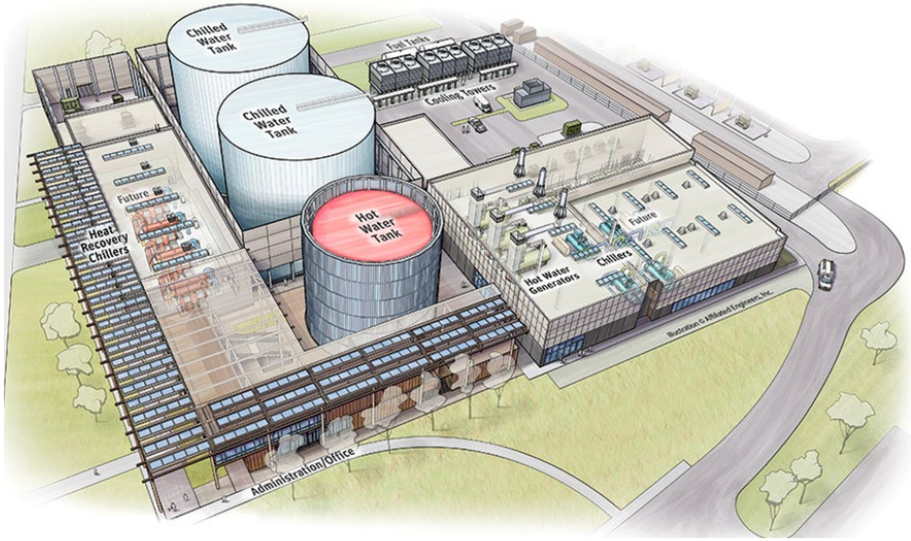
$$x_{k+1}(\xi) = A x_k(\xi) + B u_k(\xi)$$
$$x_0(\xi) = \hat{x}_t$$

Demand Charges:

$$D(\xi) \geq \beta u_k(\xi)$$
$$D(\xi) \geq \hat{D}_t$$

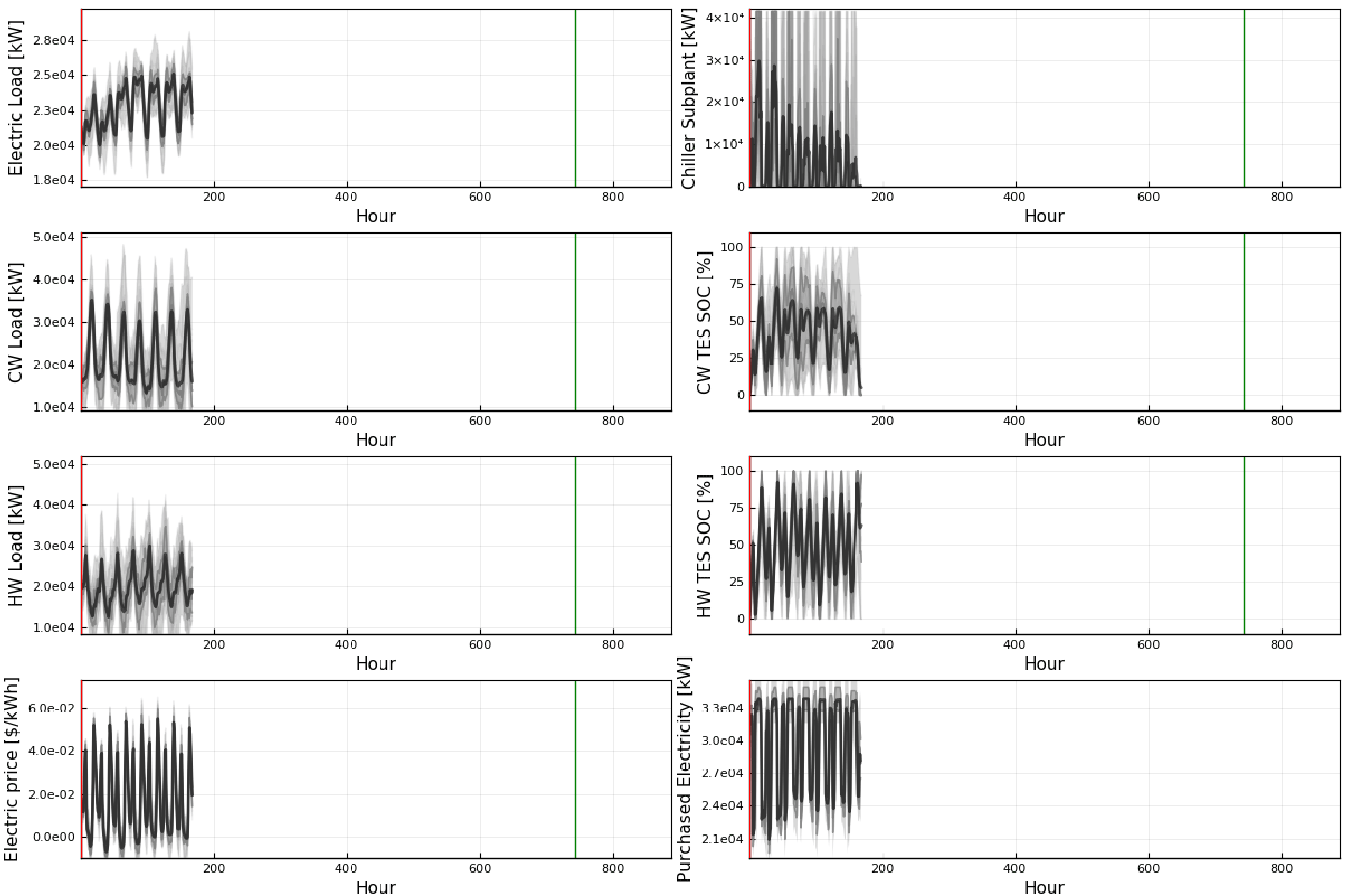
Bounds:

$$0 \leq x_k(\xi) \leq \bar{x}$$
$$\underline{u} \leq u_k(\xi) \leq \bar{u}$$
$$s_k(\xi) \geq 0$$



Closed-Loop System

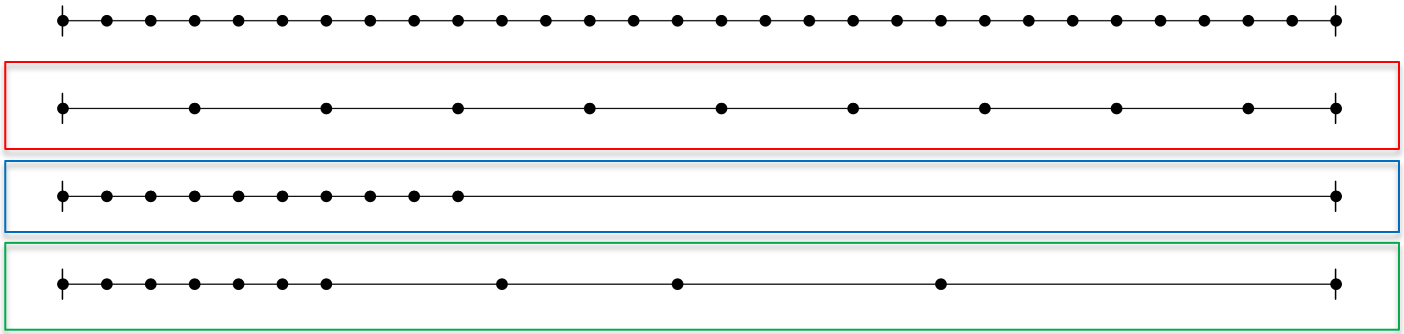
Closed-Loop Simulation (One Year)



Key: A single year-long simulation takes ∼3 hours.

Graph-Based Approximation (Diffusing Horizon MPC)

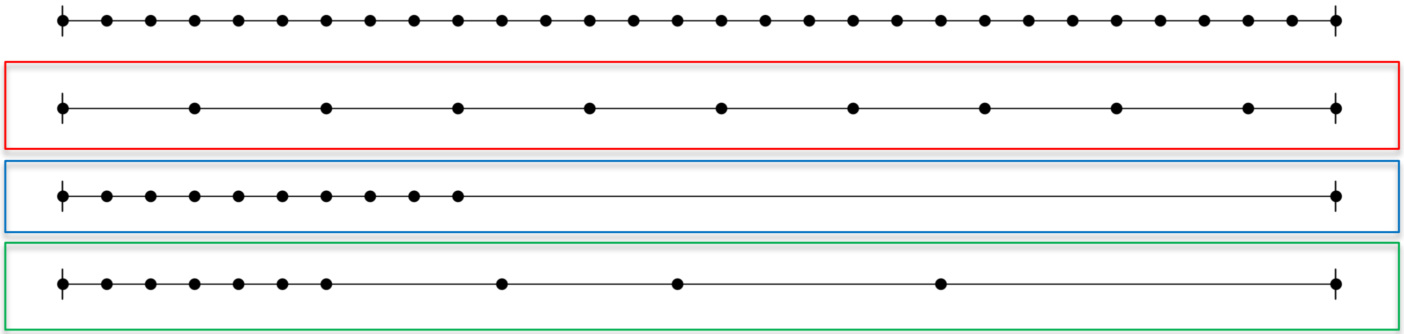
Horizon Coarsening Strategies

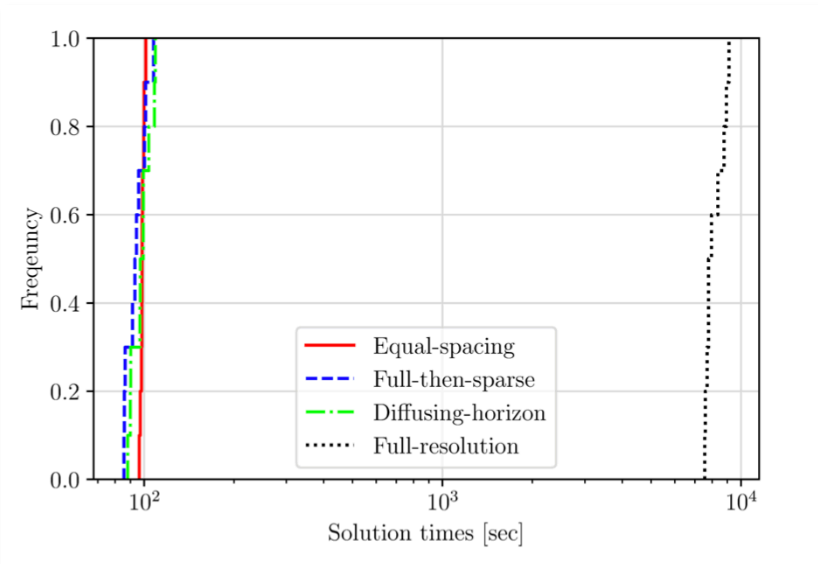


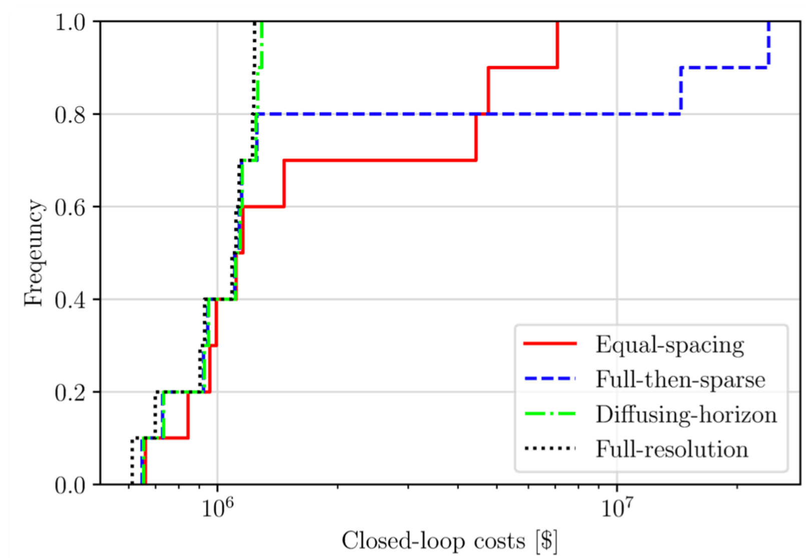
Shin, S., & Zavala, V. M. (2021). Diffusing-horizon model predictive control. *IEEE Transactions on Automatic Control*.

Graph-Based Approximation (Diffusing Horizon MPC)

Horizon Coarsening Strategies







Key: Diffusing-horizon decreases times by 2 orders of magnitude.

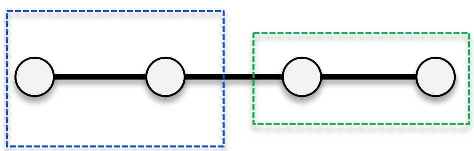
Shin, S., & Zavala, V. M. (2021). Diffusing-horizon model predictive control. *IEEE Transactions on Automatic Control*.

Overlapping Schwarz Decomposition

Full Problem

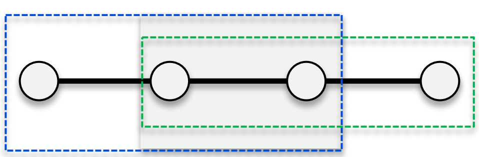
$$\begin{bmatrix} Q_{11} & Q_{12} & & \\ Q_{21} & Q_{22} & Q_{23} & \\ & Q_{32} & Q_{33} & Q_{34} \\ & & Q_{43} & Q_{44} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix}$$

Standard Partitioning



$$\begin{bmatrix} Q_{11} & Q_{12} & & \\ Q_{21} & Q_{22} & Q_{23} & \\ & Q_{32} & Q_{33} & Q_{34} \\ & & Q_{43} & Q_{44} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix}$$

Overlapping Partitioning

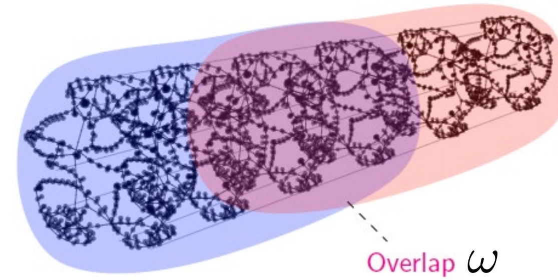


$$\begin{bmatrix} Q_{11} & Q_{12} & & \\ Q_{21} & Q_{22} & Q_{23} & \\ & Q_{32} & Q_{33} & Q_{34} \\ & & Q_{43} & Q_{44} \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} c_1 \\ c_2 \\ c_3 \\ c_4 \end{bmatrix}$$

Key: Eliminate Redundancy in Overlapping Region via Restriction

Balay, Satish, Shrirang Abhyankar, Mark Adams, Jed Brown, Peter Brune, Kris Buschelman, Lisandro Dalcin et al. "PETSc users manual." (2019).

Overlapping Schwarz for QP



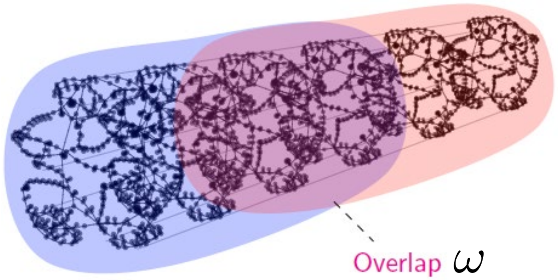
Consider graph-structured problem:

$$\min_z \frac{1}{2} z^T Q z + c^T z$$

$Q \succ 0$ embeds graph structure $G(V, E)$

Shin, S., Zavala, V. M., & Anitescu, M. (2018). Decentralized schemes with overlap for solving graph-structured optimization problems. *arXiv preprint arXiv:1810.00491*.

Overlapping Schwarz for QP



Consider graph-structured problem:

$$\min_z \frac{1}{2} z^T Q z + c^T z$$

$Q \succ 0$ embeds graph structure $G(V, E)$

Partition domain V into V_k, $k = 1, ..., K$ and construct overlapping domains:

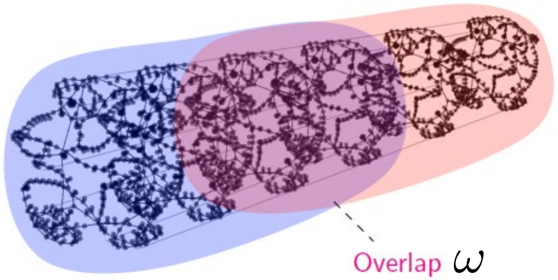
$$V_k^\omega := \{v \in V \mid d(v, V_k) \leq \omega\}$$

ω is size of overlap

Key: $\omega = 0$ is decentralized and $\omega = |V|$ is centralized

Shin, S., Zavala, V. M., & Anitescu, M. (2020). Decentralized schemes with overlap for solving graph-structured optimization problems. *IEEE Transactions on Control of Network Systems*, *7*(3), 1225-1236.

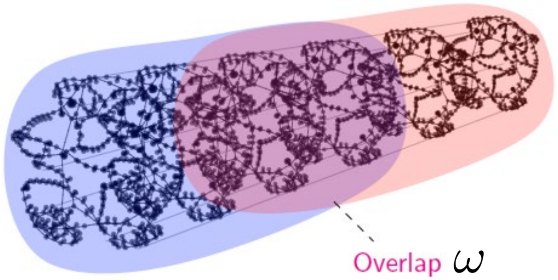
Overlapping Schwarz for QP



Overlapping Schwarz Scheme is:

$$\boxed{z_k^{\ell+1} = R_k \operatorname*{argmin}_{z_k} \frac{1}{2} z_k^T Q_k z_k + (c_k - Q_{-k} z_{V_{-k}}^{\ell})^T z_k} \quad \text{for } k = 1, ..., K$$

Overlapping Schwarz for QP



Overlapping Schwarz Scheme is:

$$z_k^{\ell+1} = R_k \operatorname*{argmin}_{z_k} \frac{1}{2} z_k^T Q_k z_k + (c_k - Q_{-k} z_{V_{-k}}^{\ell})^T z_k \quad \text{for } k = 1, \ldots, K$$

Theorem: The scheme converges linearly:

$$\left\| z^{\ell} - z^* \right\|_{\infty} \leq \alpha^{\ell} \left\| z^0 - z^* \right\|_{\infty}$$

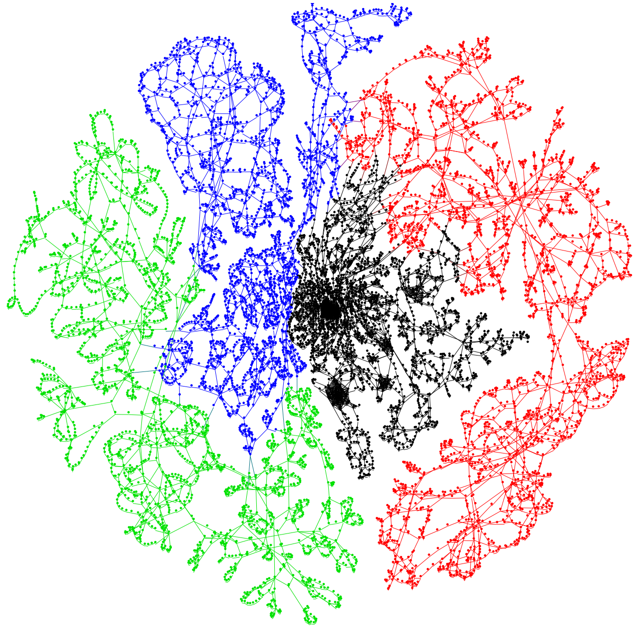
with convergence rate:

$$0 \leq \alpha \leq \frac{\delta_1}{\lambda_{\min}} \left(\frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}} \right)^{(\omega+1)/\delta_2}$$

Key: Convergence rate improves *exponentially* with ω.

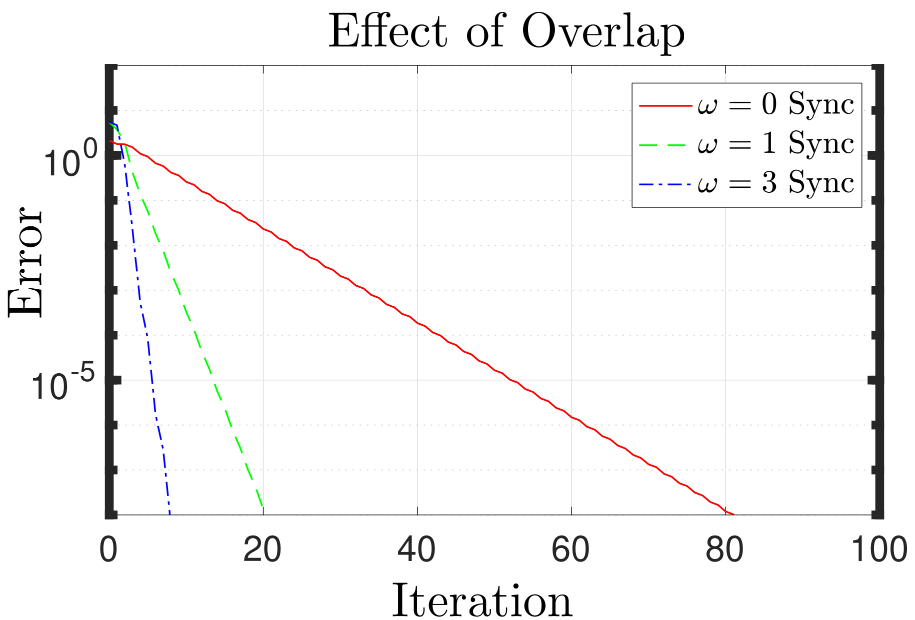
State Estimation

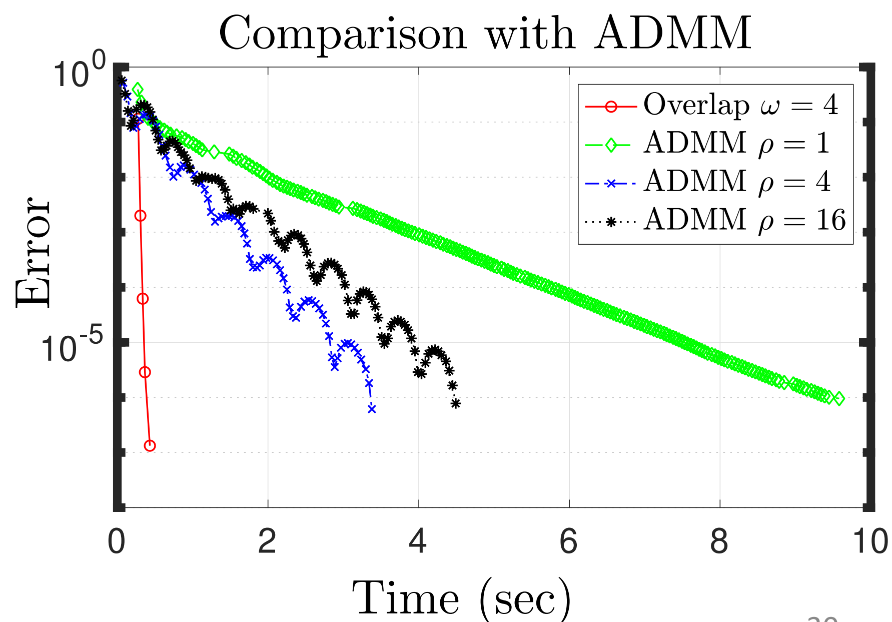
9,241 Node Power Network (Pegasus):



$$\min_{\delta,P} \sum_{i\in V}\left(\frac{\delta_i-\delta_i^m}{\sigma_{\delta_i}}\right)^2+\sum_{\{i,j\}\in E}\left(\frac{P_{ij}-P_{ij}^m}{\sigma_{P_{ij}}}\right)^2$$

$$\text{s.t. } P_{ij}=y_{ij}(\delta_i-\delta_j),\quad \{i,j\}\in E$$

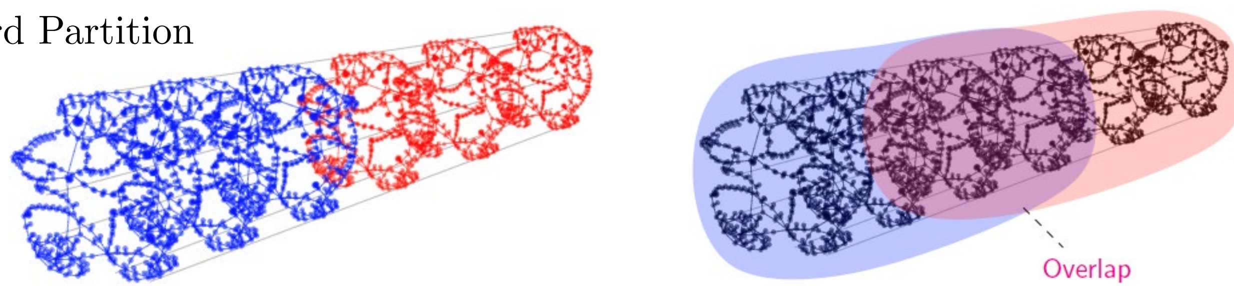




Overlapping Schwarz for NLP

$$\min_{\{x_i\}_{i \in V}} \sum_{i \in V} f_i(\{x_j\}_{j \in N_G[i]})$$
$$\text{s.t.} \;\; c_i(\{x_j\}_{j \in N_G[i]}) \geq 0, \;\; i \in V$$

Standard Partition



Overlapping Partition

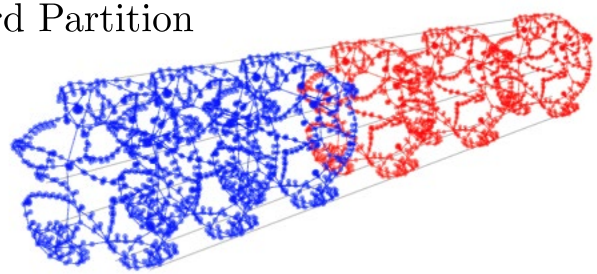
$$\{x_i^{(\ell+1)}\}_{i \in V_k} = R_k \operatorname{argmin} f(\{x_i\}_{i \in V_k^\omega}; \{x_i^{(\ell)}\}_{i \notin V_k^\omega})$$
$$\text{s.t.} c(\{x_i\}_{i \in V_k^\omega}; \{x_i^{(\ell)}\}_{i \notin V_k^\omega}) \geq 0$$

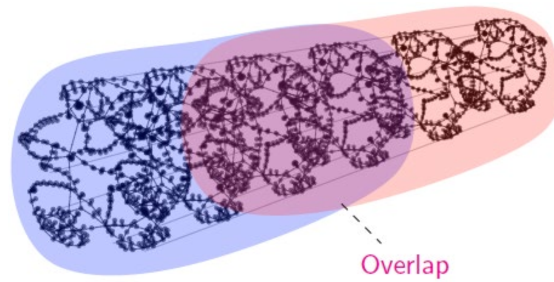
Shin, S. (2021). *Graph-Structured Nonlinear Programming: Properties and Algorithms* (Doctoral dissertation, The University of Wisconsin-Madison).

Overlapping Schwarz for NLP

$$\min_{\{x_i\}_{i\in V}} \sum_{i\in V} f_i(\{x_j\}_{j\in N_G[i]})$$
$$\text{s.t.}\;\; c_i(\{x_j\}_{j\in N_G[i]}) \geq 0,\; i\in V$$

Standard Partition





Overlapping Partition

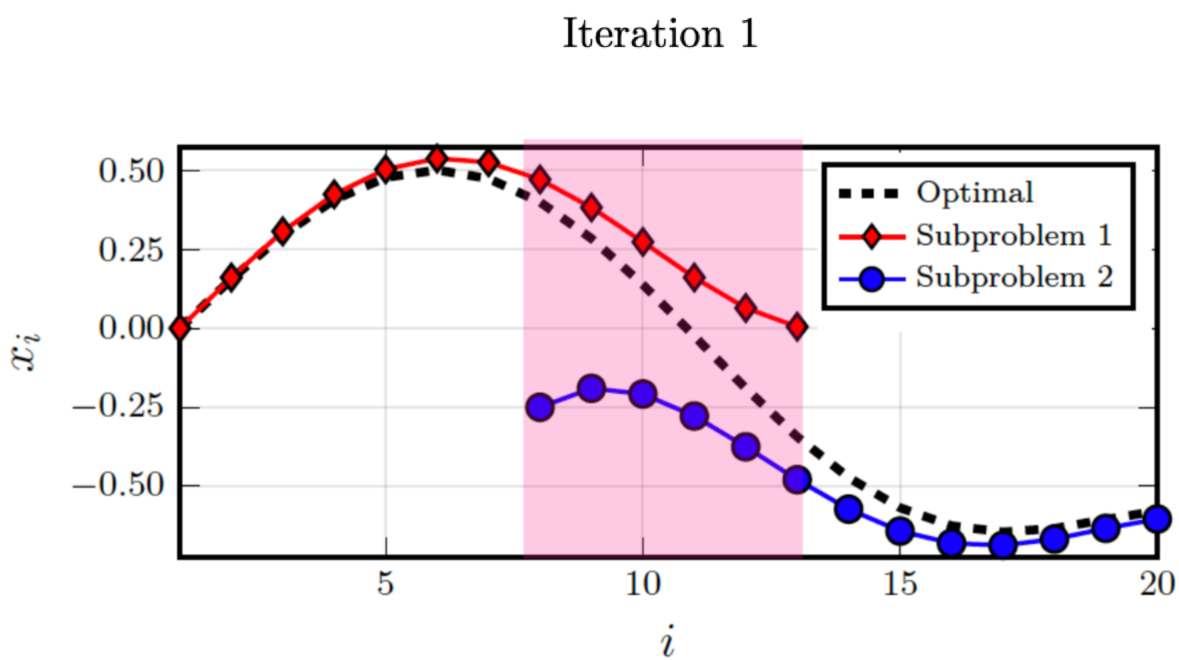
$$\{x_i^{(\ell+1)}\}_{i\in V_k} = R_k \operatorname{argmin} f(\{x_i\}_{i\in V_k^\omega}; \{x_i^{(\ell)}\}_{i\notin V_k^\omega})$$
$$\text{s.t.}\, c(\{x_i\}_{i\in V_k^\omega}; \{x_i^{(\ell)}\}_{i\notin V_k^\omega}) \geq 0$$

Key: - Linear Local Convergence (Rate Improves *Exponentially* with Overlap)

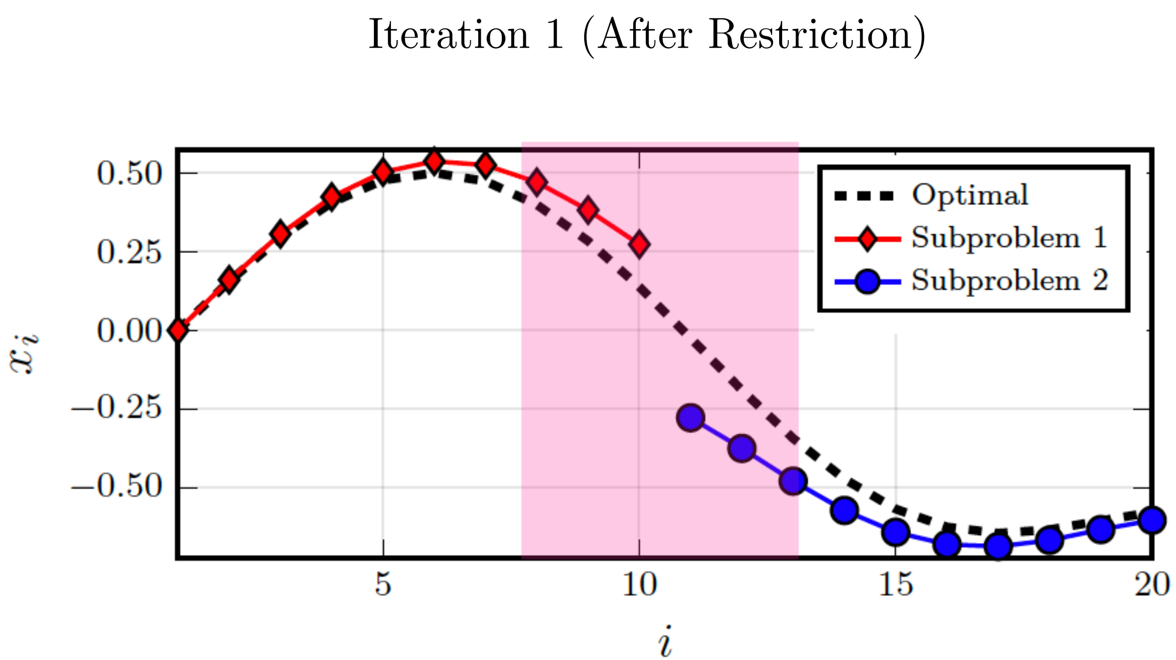
- *Exponential Decay of Sensitivity* (EDS) Guarantees Convergence.
- Schwarz Can be Used Within Interior-Point Solver (as Linear/QP Solver)

Shin, S. (2021). *Graph-Structured Nonlinear Programming: Properties and Algorithms* (Doctoral dissertation, The University of Wisconsin-Madison).

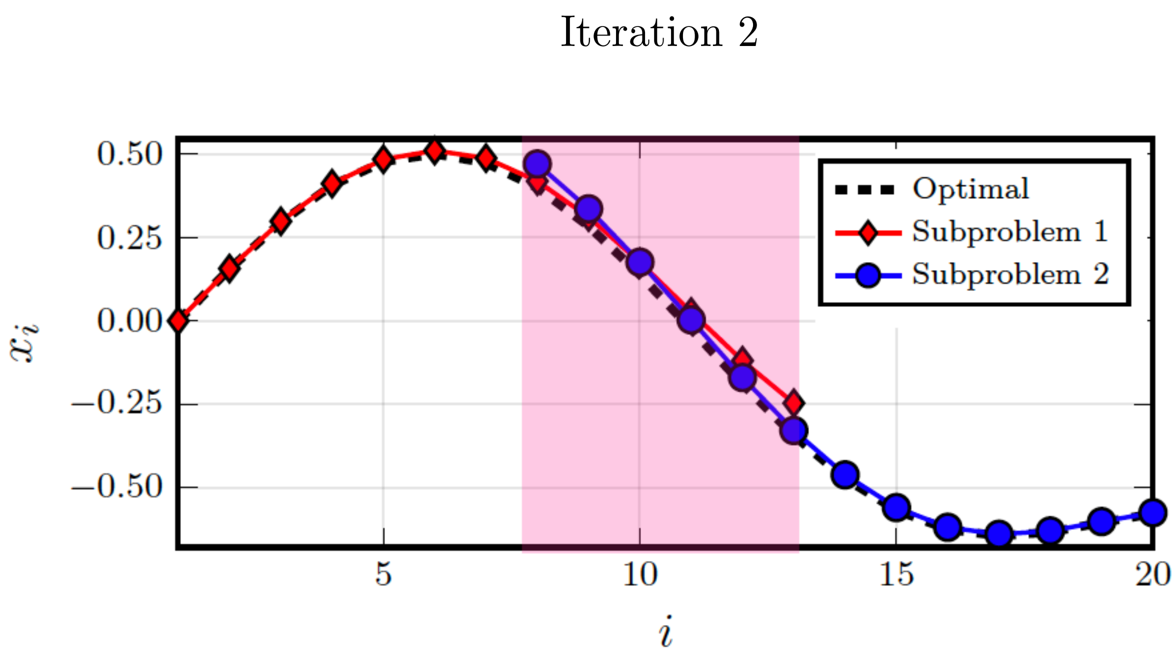
EDS Guarantees Convergence



EDS Guarantees Convergence



EDS Guarantees Convergence



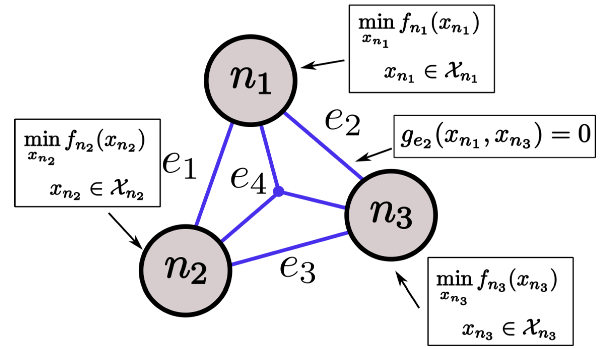
Software

Plasmo.jl

https://github.com/zavalab/Plasmo.jl

$$\min_{\{x_i\}_{i \in V}} \sum_{i \in V} f_i(\{x_j\}_{j \in N_G[i]})$$
$$\text{s.t.} \quad c_i(\{x_j\}_{j \in N_G[i]}) \geq 0, \; i \in V$$

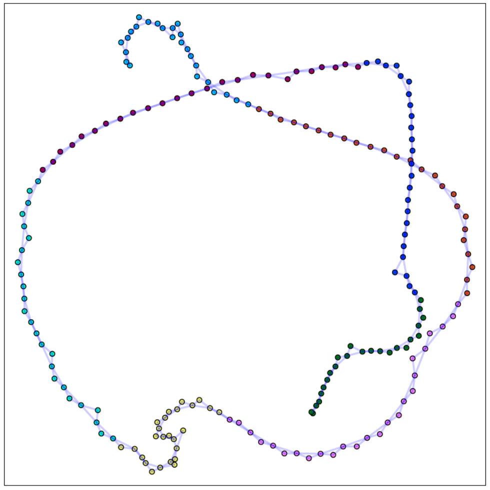
$\iff$



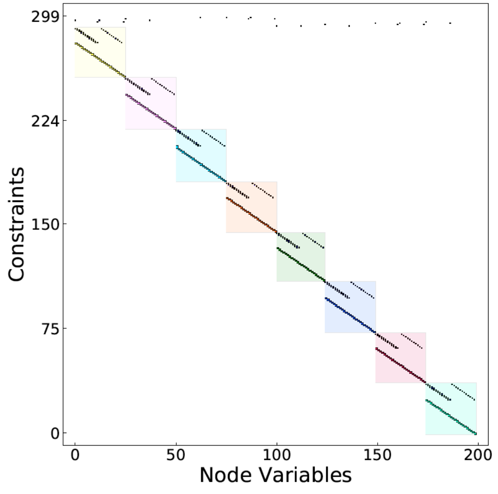
Graph Building

```julia
using Plasmo
using Plots
using Ipopt

T = 100  #number of time points
d = sin.(1:T)  #disturbance

#Create an OptiGraph
graph = OptiGraph()

#Add nodes for states and controls
@optinode(graph,state[1:T])
@optinode(graph,control[1:T-1])

#Add state variables
for (i,node) in enumerate(state)
    @variable(node,x)
    @constraint(node, x >= 0)
    @objective(node,Min,x^2)
end

#Add control variables
for node in control
    @variable(node,u)
    @constraint(node, u >= -1000)
    @objective(node,Min,u^2)
end

#Initial condition
n1 = state[1]
@constraint(n1,n1[:x] == 0)

#Dynamic coupling
@linkconstraint(graph, [t = 1:T-1],state[t+1][:x] == state[t][:x] +
                control[t][:u] + d[t])

#Optimize with Ipopt
ipopt = Ipopt.Optimizer
optimize!(graph,ipopt)

#Plot result structure
plt_graph4 = plot(graph,
layout_options = Dict(:tol => 0.1,:iterations => 500),
linealpha = 0.2,markersize = 6)
```

Graph Structure



Jacobian Structure



Plasmo.jl

```
using Plasmo, Ipopt
```

- Plasmo.jl provides a **user-friendly** syntax similar to JuMP.jl
- Each node embeds a JuMP.jl optimization model
- Node subproblems linked using hyperedges

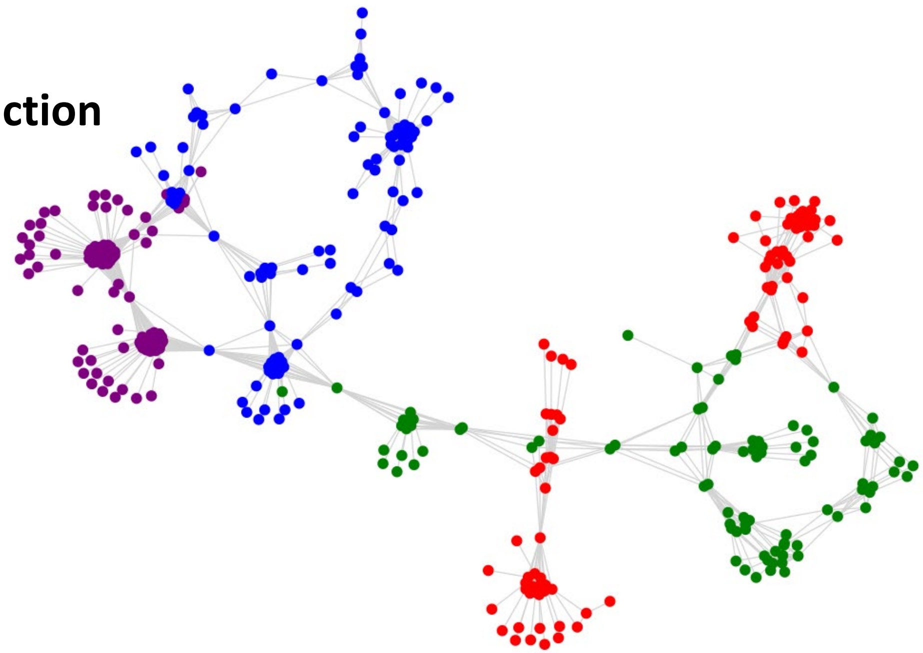
Jalving, J., Shin, S., & Zavala, V. M. (2020). A Graph-Based Modeling Abstraction for Optimization: Concepts and Implementation in Plasmo.jl.

Graph Visualization

System

$\Longleftrightarrow$

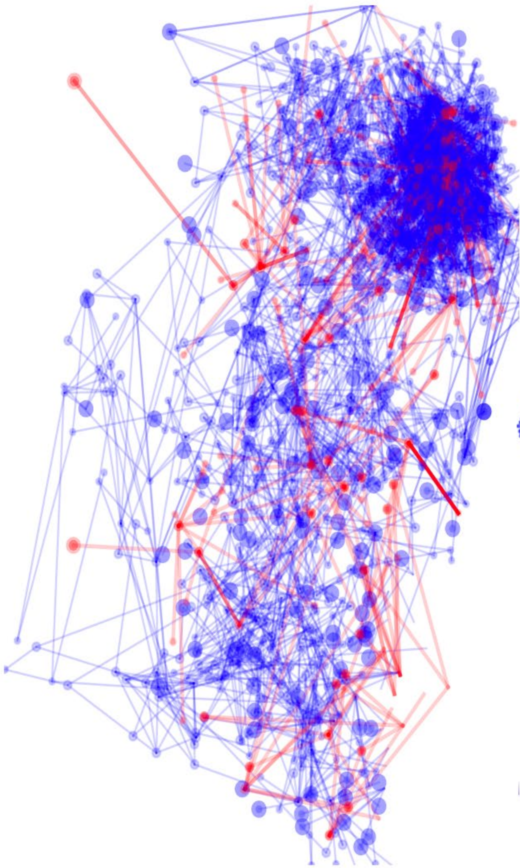
Graph Abstraction



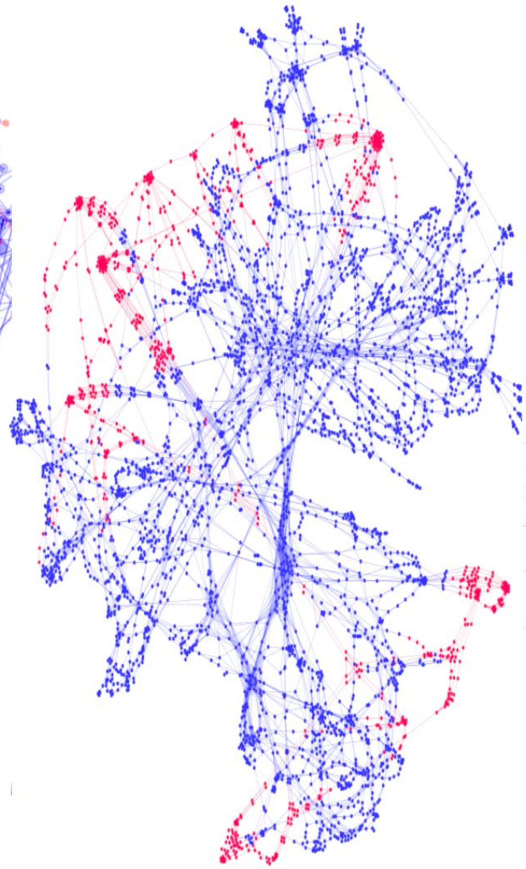
- Graph Abstraction Provides Insights (e.g., Connectivity, Vulnerability)

Graph Visualization

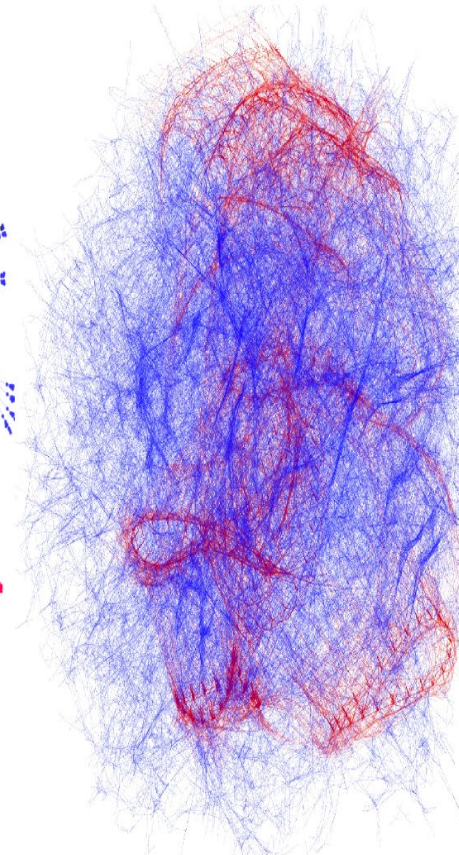
- Plasmo.jl facilitates Graph Manipulation, Visualization, and Decomposition



Gas-Electric Network



Space Unfolding

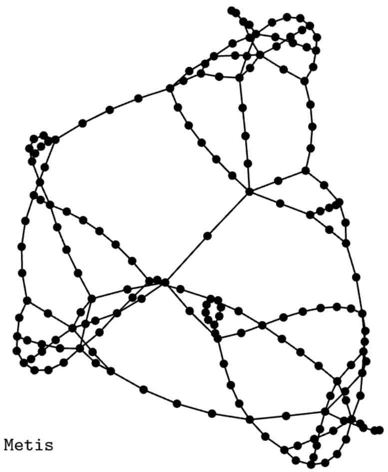


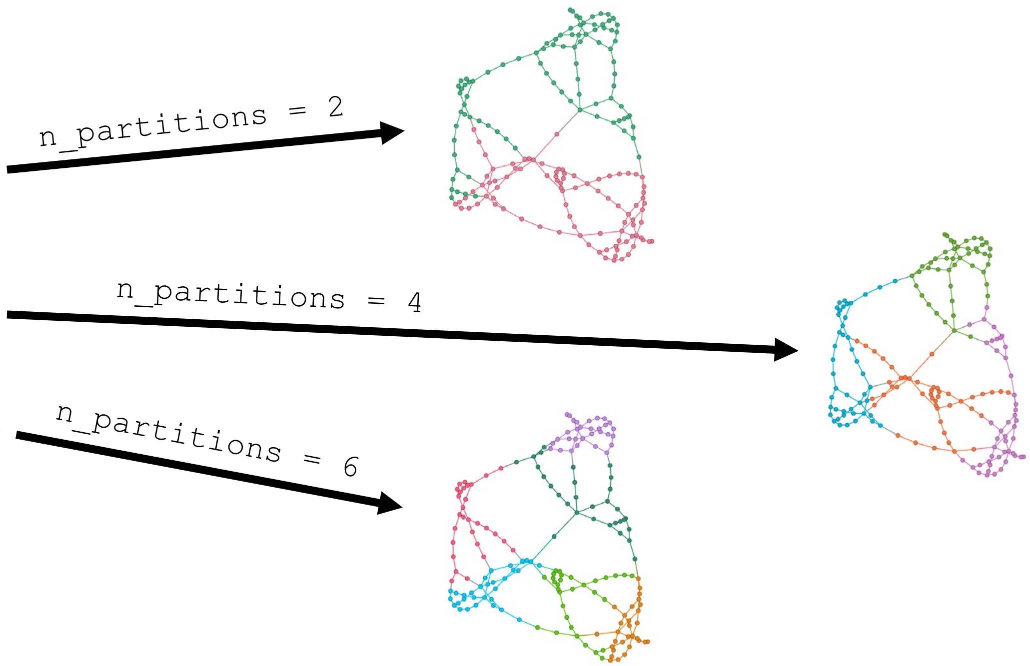
100,000 nodes
300,000 edges

Space-Time Unfolding

Jalving, J., Shin, S., & Zavala, V. M. (2020). A Graph-Based Modeling Abstraction for Optimization: Concepts and Implementation in Plasmo.jl.

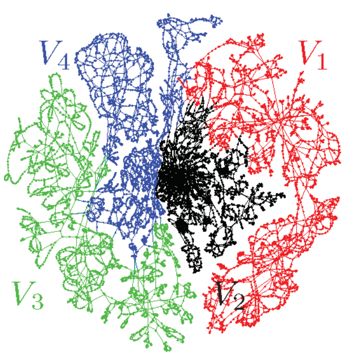
Graph Partitioning



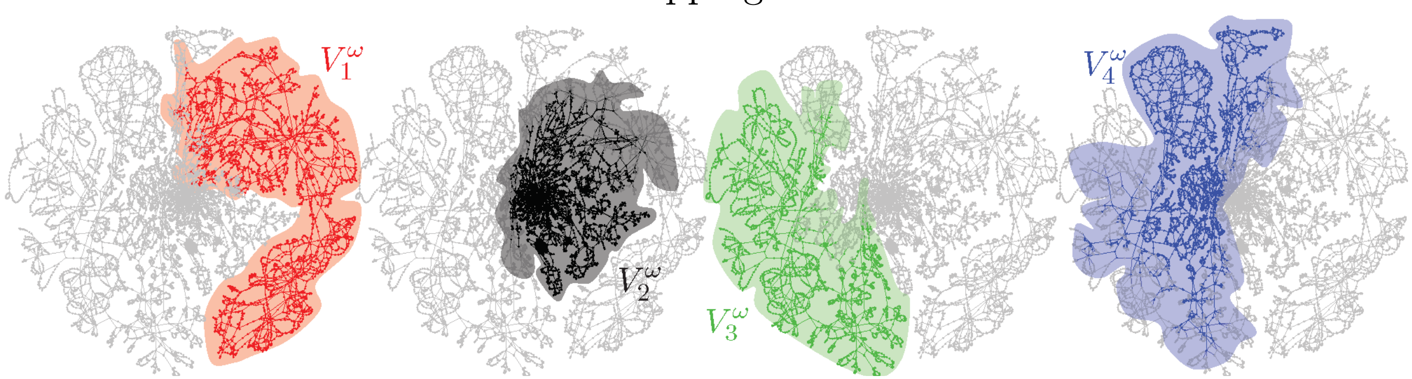



```julia
using Plasmo, Metis

g = OptiGraph()
....

partition_vector = Metis.partition(
             simplegraph, n_partitions, alg = :KWAY)
....
partition = Partition(hypergraph,
              partition_vector, hyper_map)

apply_partition!(g, partition)
```

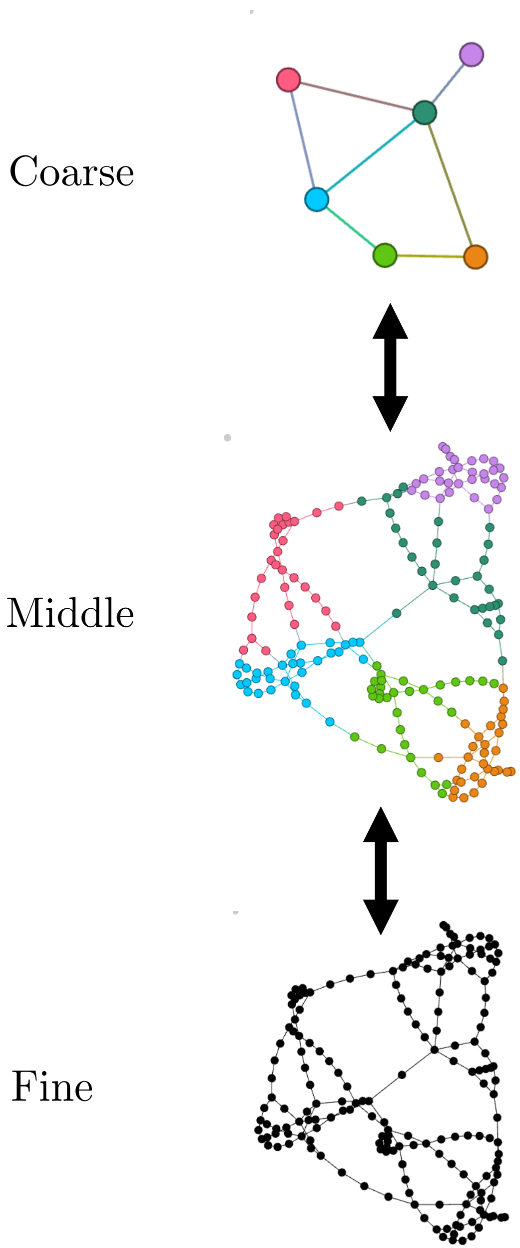
Standard Partition

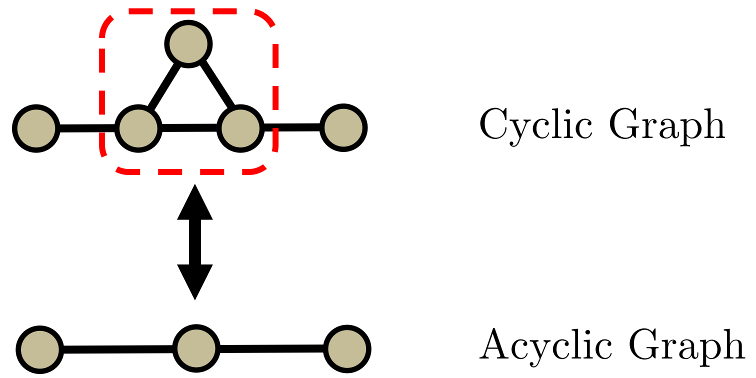


Overlapping Partition

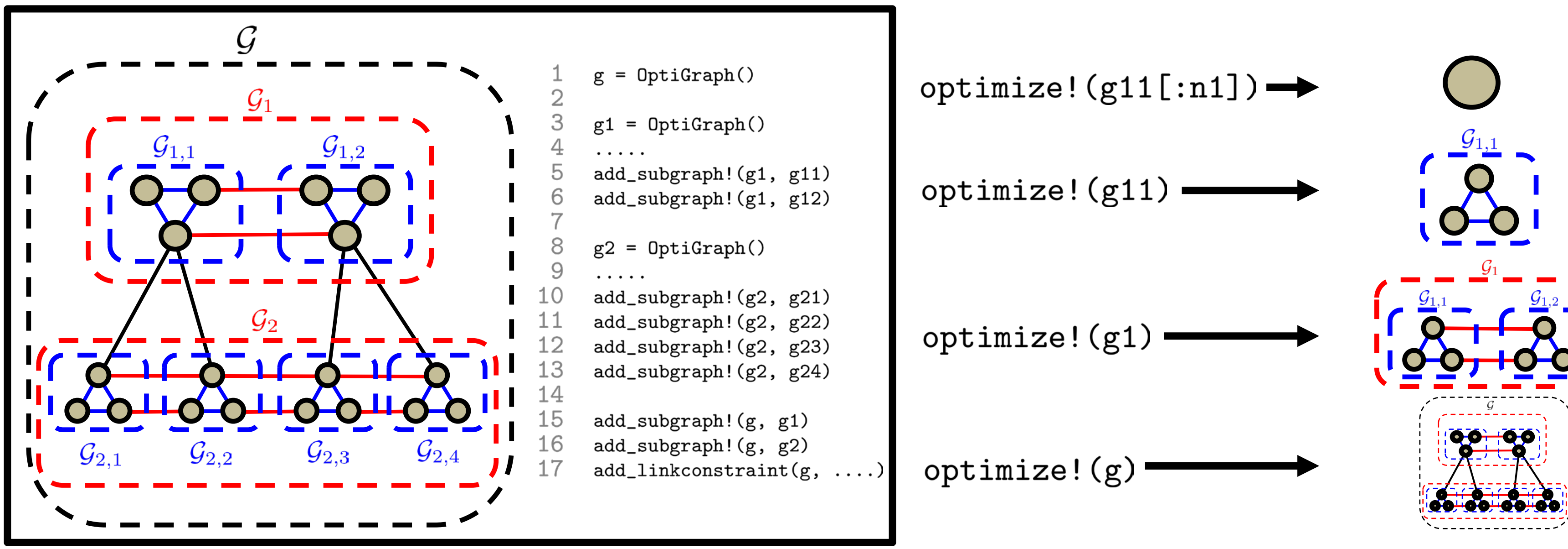


Graph Aggregation/Coarsening





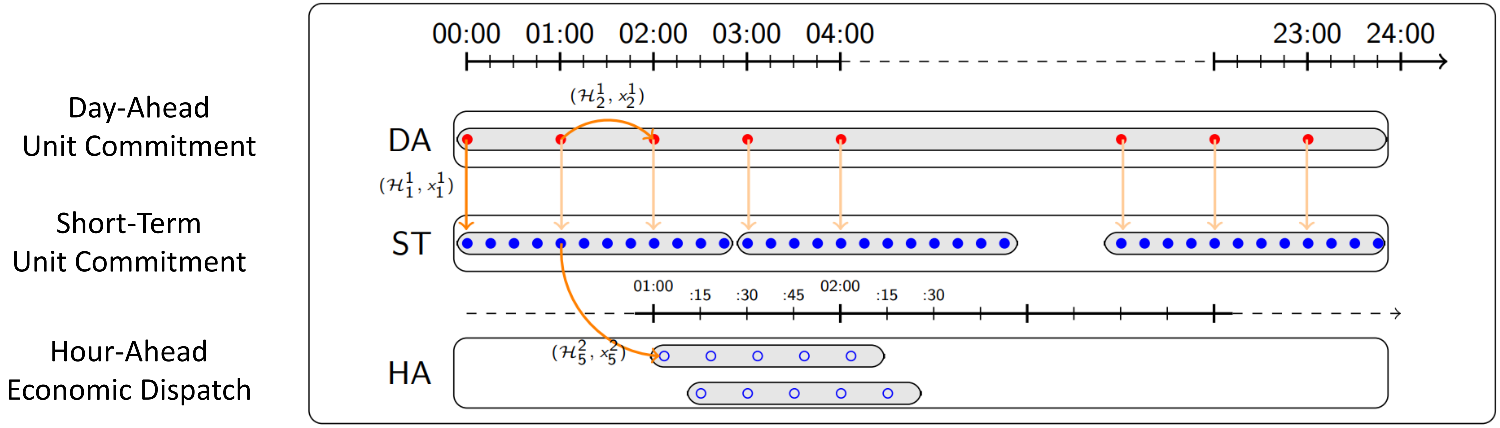
Hierarchical Graphs

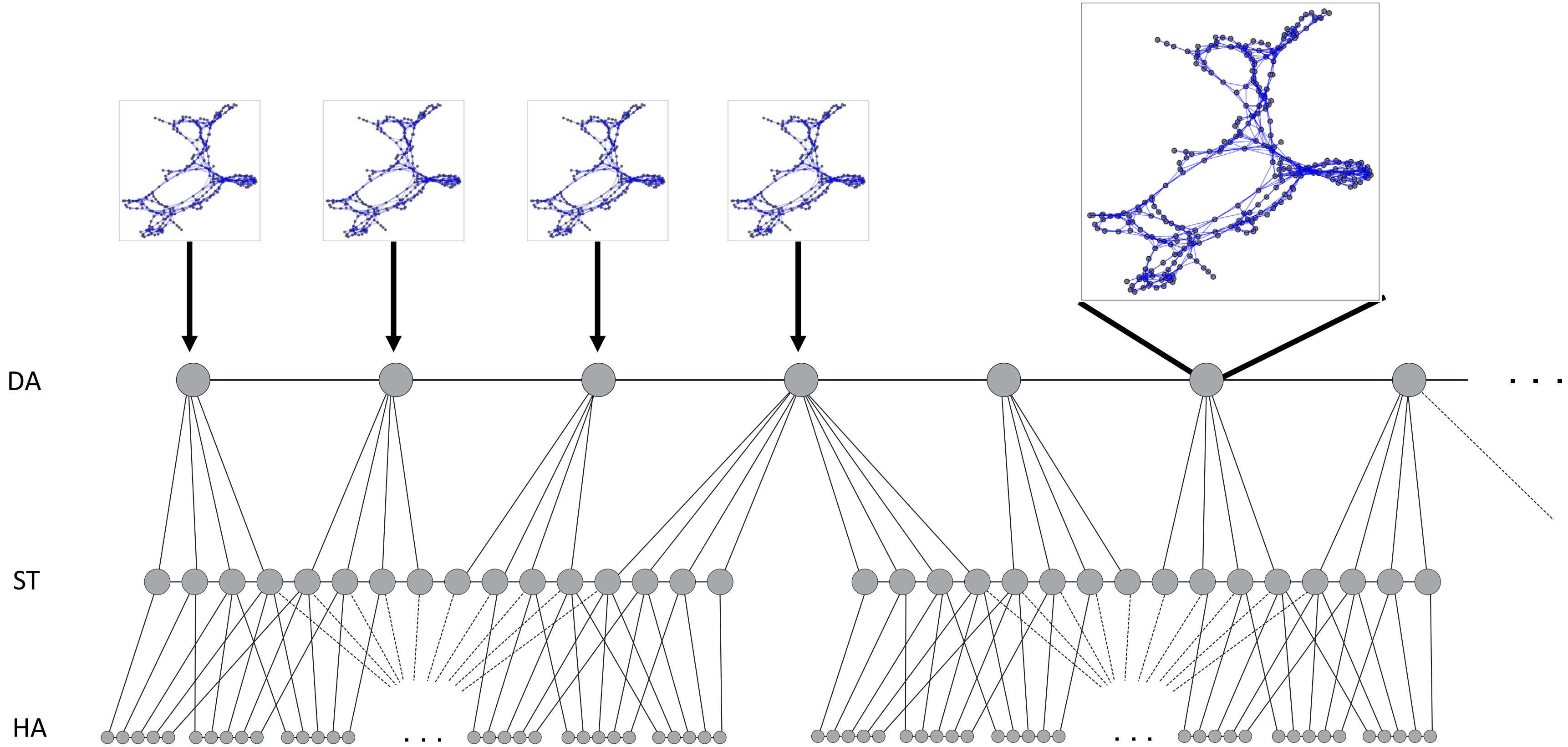


- Nodes can embed full graphs
- Enables **modular & hierarchical** modeling
- Can **solve subgraphs** all-at-once, individually, or as communities/partitions

Cole, D. L., Pecci, F., Guerra, O. J., Gangammanavar, H., Jenkins, J. D., & Zavala, V. M. (2025). Graph-Based Modeling and Decomposition of Hierarchical Optimization Problems. *arXiv preprint arXiv:2501.02098*.

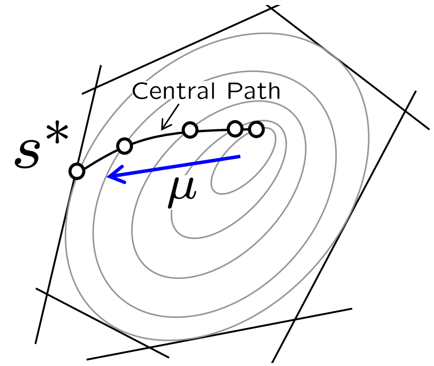
Multi-Scale MPC as Hierarchical Graphs



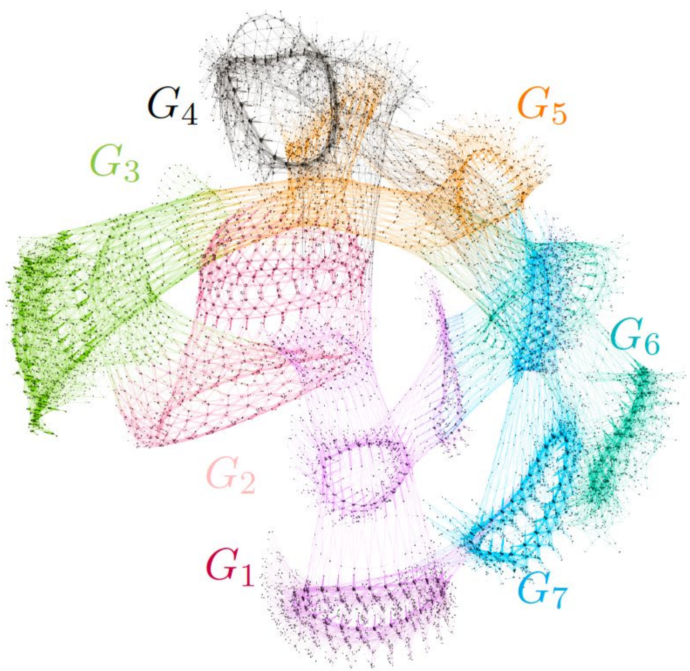


MadNLP.jl https://github.com/zavalab/MadNLP.jl

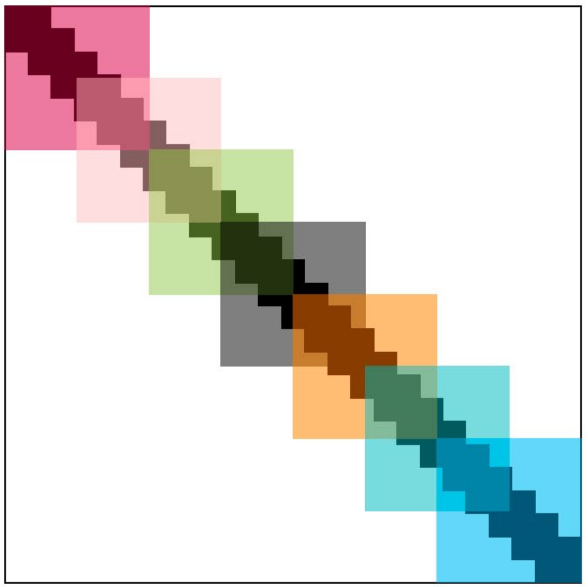
$$\min_{\{x_i\}_{i\in V}} \sum_{i\in V} f_i(\{x_j\}_{j\in N_G[i]})$$
$$\text{s.t. } c_i(\{x_j\}_{j\in N_G[i]}) \geq 0,\ i \in V$$



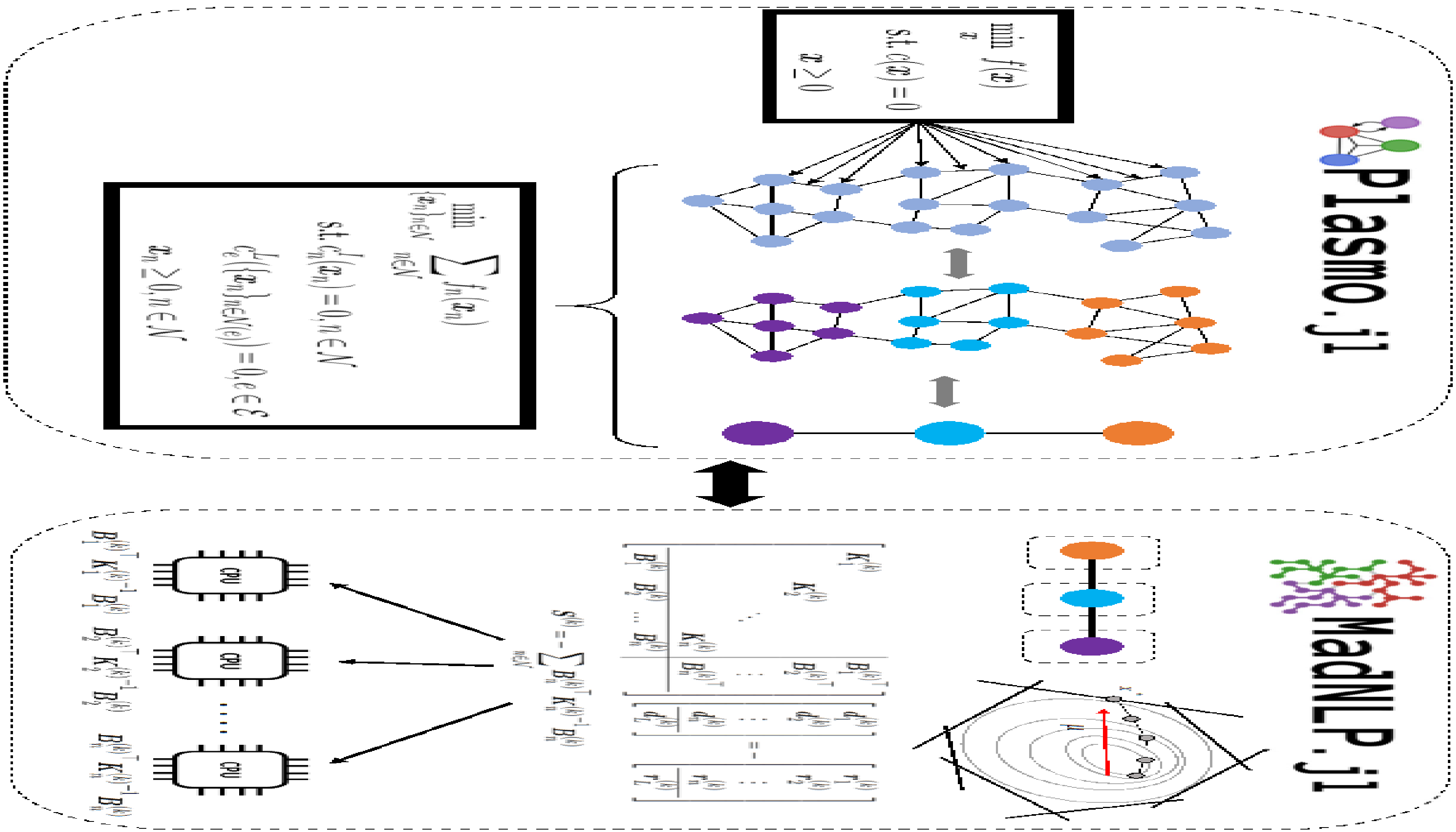
- MadNLP.jl is a Julia-Based Interior-Point Optimizer
- Exploits Graph Structures at Linear Algebra Level (e.g., Schwarz, Schur)



Problem Graph


Linear Algebra (KKT) System

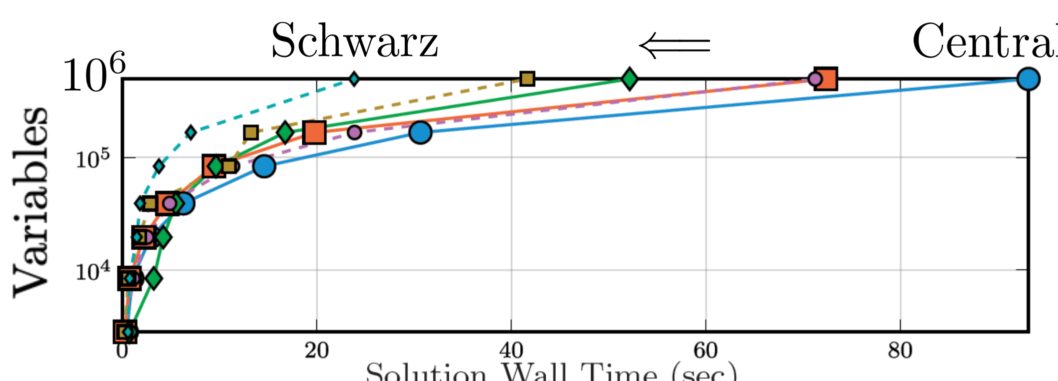
Plasmo.jl + MadNLP.jl



Plasmo.jl + MadNLP.jl

- Optimal Control of Gas Network (NLP with 10^6 Variables)
- Schwarz Decomposition Accelerates Solution by 300%

$$
\begin{aligned}
\min_{\substack{\rho,\varphi^a,\varphi^-,\\ \alpha,s,d\in\mathbb{R}}} \; & \sum_{t\in\mathcal{T}} \left(\sum_{(i,j)\in\mathcal{C}} \gamma P^a_{ijt} + \sum_{i\in\mathcal{R}} c_{it} s_{it} - \sum_{i\in D} c_{it} d_{it} \right) \\
\text{s.t.} \; & \sum_{j\in\mathcal{N}(i)} f^a_{ijt} = \sum_{j\in\mathcal{R}(i)} s_{jt} - \sum_{j\in\mathcal{D}(i)} d_{jt}, i\in\mathcal{N}, t\in\mathcal{T} \\
& \rho_i^{\min} \le \rho_{it} \le \rho_i^{\max},\; i\in\mathcal{N},\; t\in\mathcal{T} \\
& \rho_{it}^2 - \rho_{jt}^2 = -\frac{\lambda L}{D}\varphi^a_{ijt}|\varphi^a_{ijt}|,\; (i,j)\in\mathcal{P}, t\in\mathcal{T} \\
& \hat{L}(\dot\rho_{jt} + \dot\rho_{it}) = -4\varphi^-_{ijt},\; (i,j)\in\mathcal{P},\; t\in\mathcal{T} \\
& f^a_{ijt}(\rho_{it} - \rho_{jt}) \le 0,\; (i,j)\in\mathcal{C},\; t\in\mathcal{T} \\
& \rho_{jt} = \alpha_{ijt}\rho_{it},\; (i,j)\in\mathcal{C},\; t\in\mathcal{T} \\
& P^a_{ijt} \le P^{\max}_{ij},\; (i,j)\in\mathcal{C},\; t\in\mathcal{T} \\
& -f^a_{ij} \le f^a_{ijt} \le f^a_{ij},\; (i,j)\in\mathcal{C},\; t\in\mathcal{T},
\end{aligned}
$$

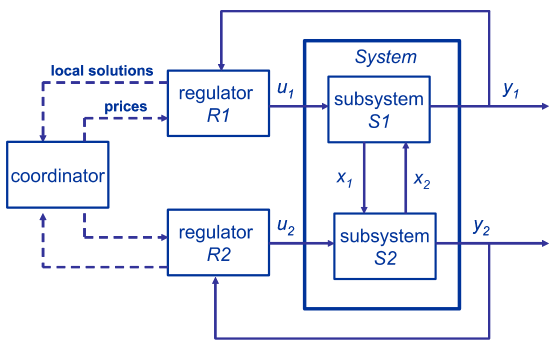


Cyber-Physical Systems

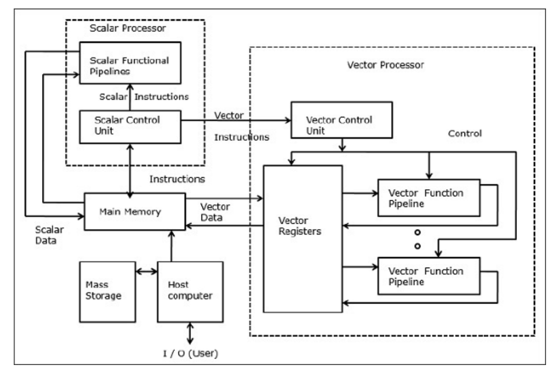
Modeling Cyber-Physical Systems

What is a General Mathematical Abstraction?

Control Architectures



Computing Architectures



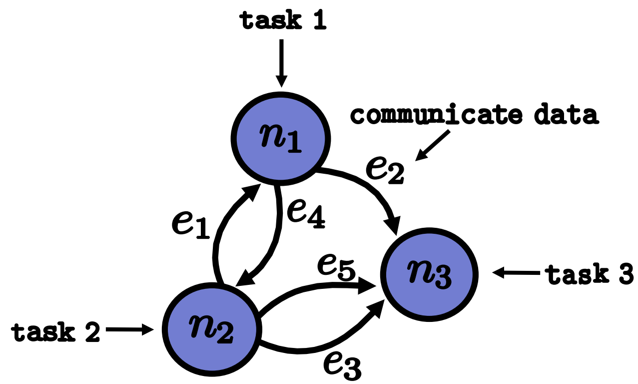
- Physical Layer are Unit Operations, Materials
- Cyber Layer are Sensors, Data, Controls, Computers...
- Cyber Layer Makes Decisions that Run Physical Layer
- **Issues**: Memory/Power, Communication, Latency, Security/Privacy, Resilience,...

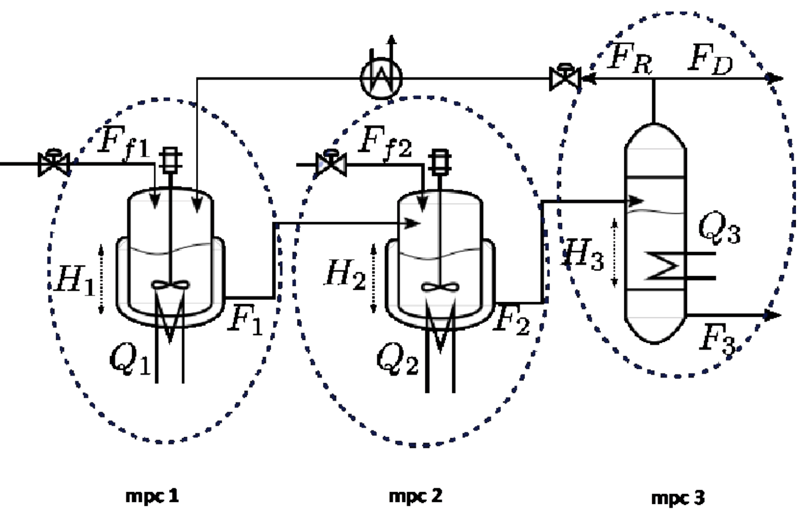
Scattolini, R. (2009). Architectures for distributed and hierarchical model predictive control–a review. *Journal of process control*, *19*(5), 723-731.

Cyber-Physical Systems as Computing Graphs

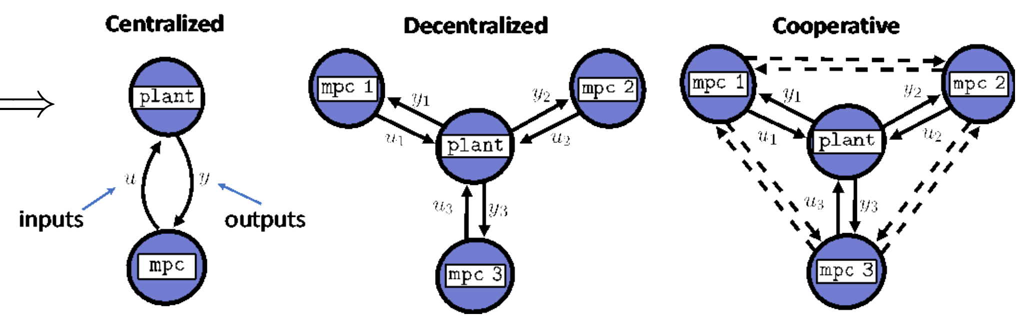
- Nodes are Computing Tasks
- Edges are Communication Channels
- Graph Captures Communication Topology

$\Longleftrightarrow$



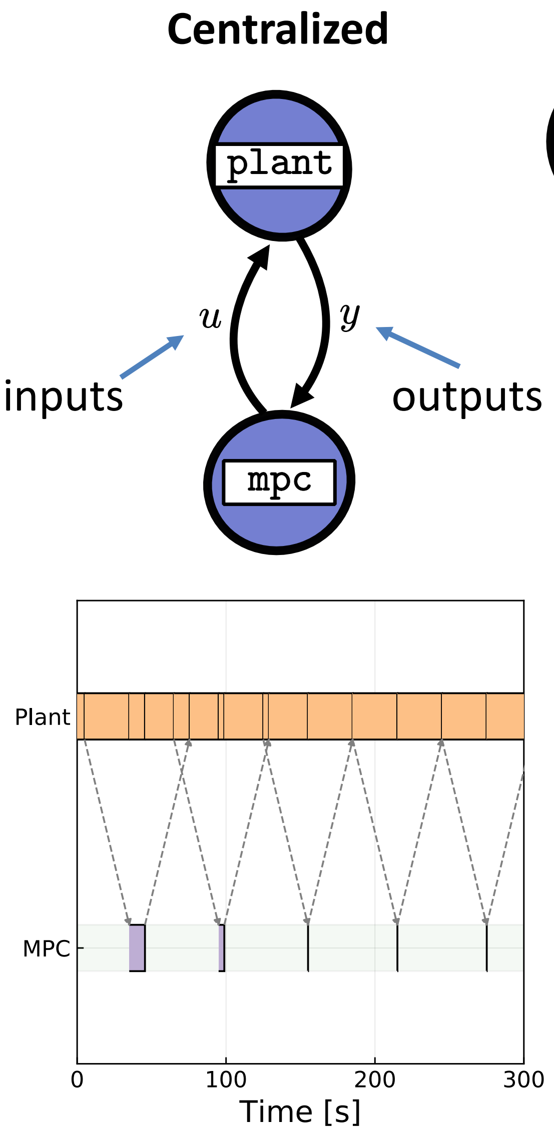


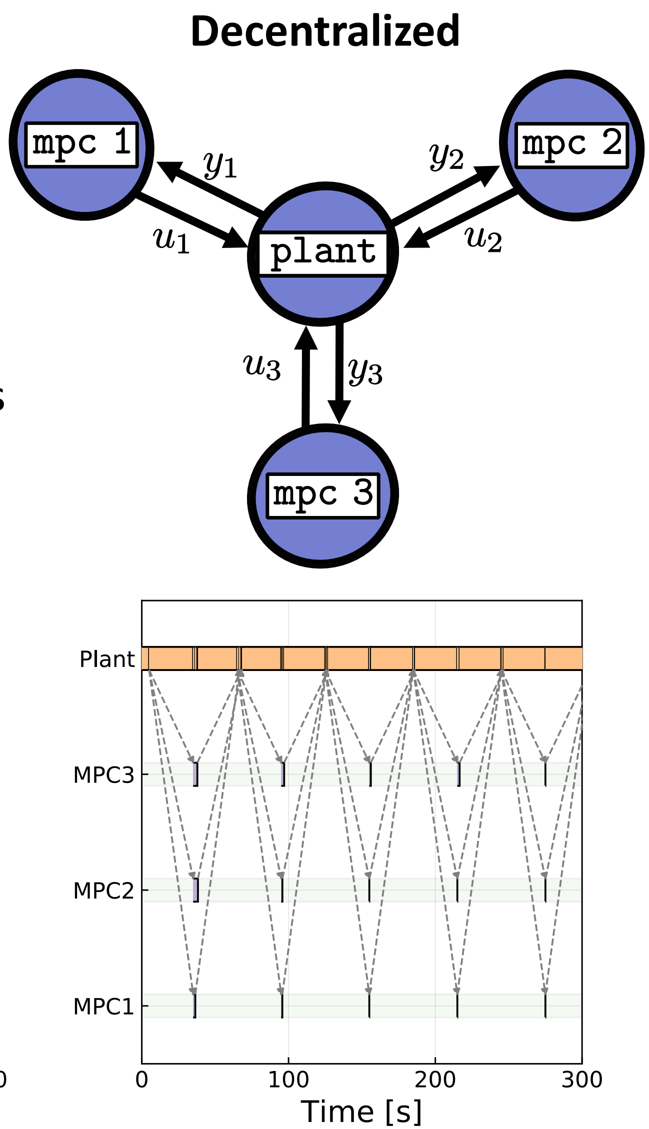
$\Longleftrightarrow$

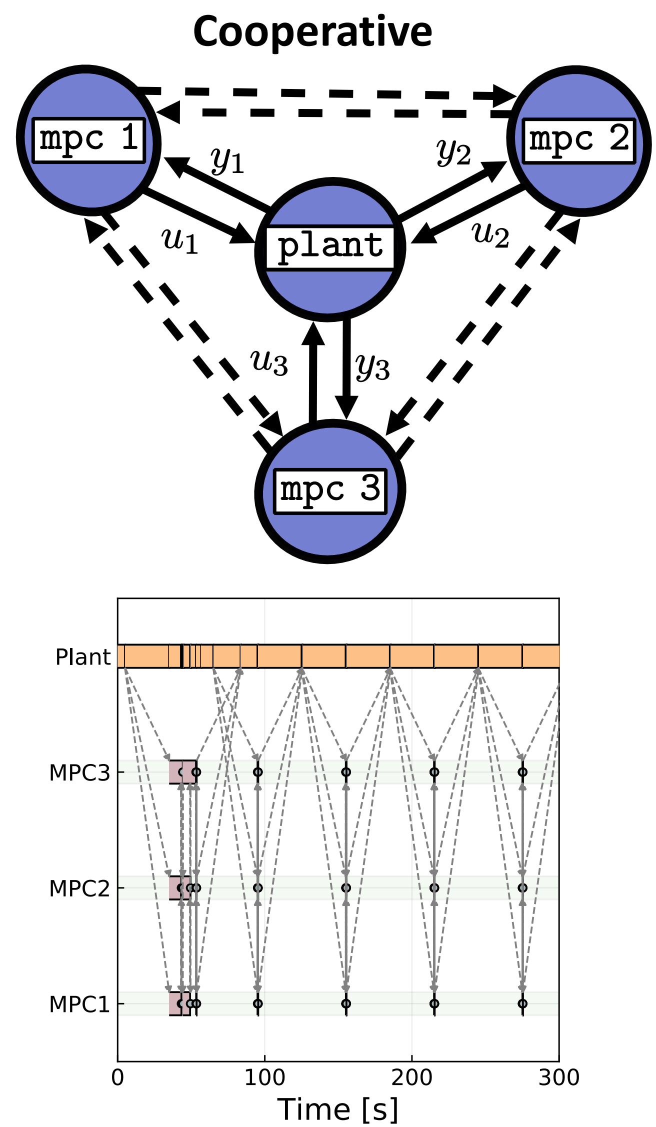


Jalving, J., Cao, Y., & Zavala, V. M. (2019). Graph-based modeling and simulation of complex systems. *Computers & Chemical Engineering*, *125*, 134-154.

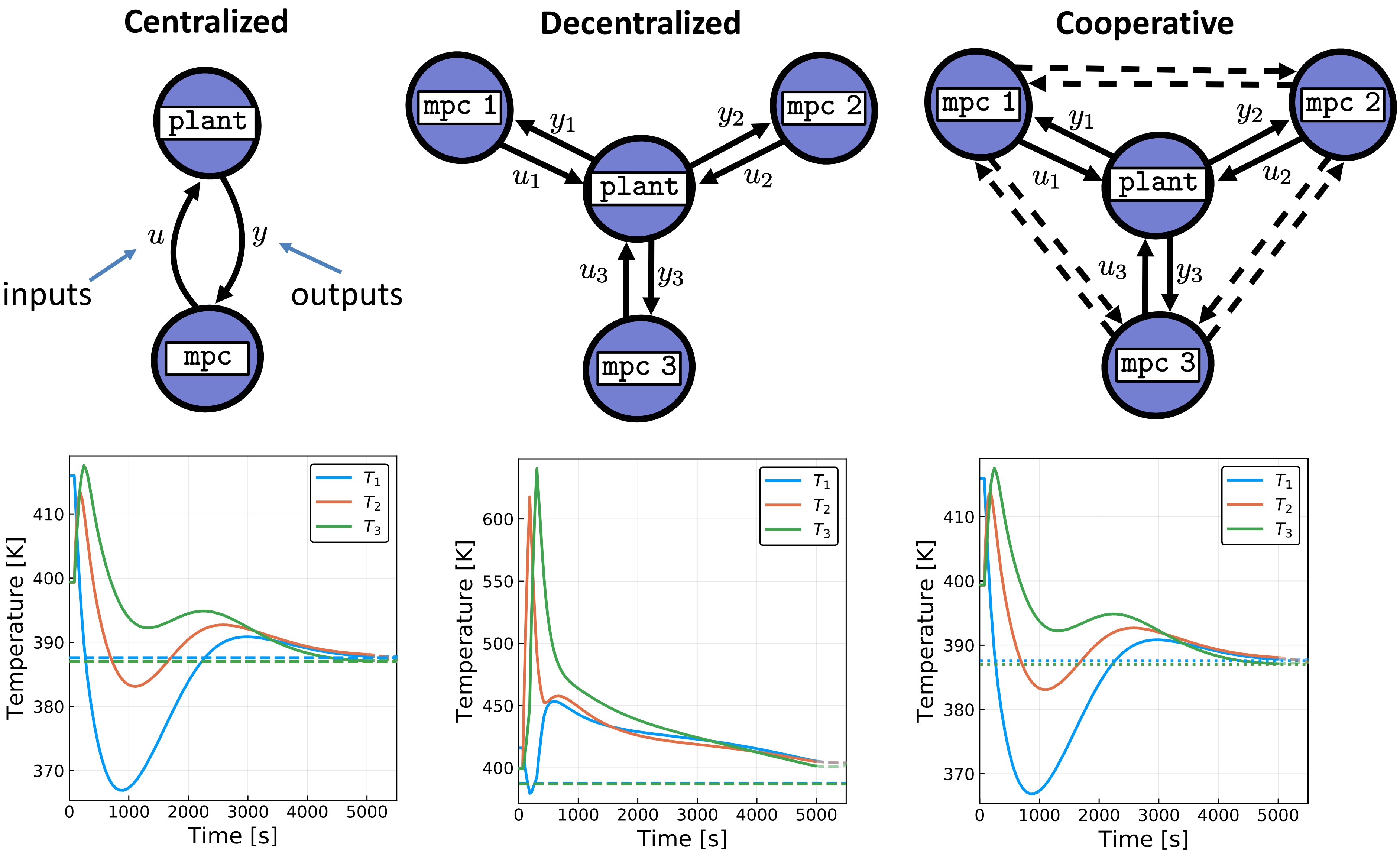
Simulating MPC Architectures





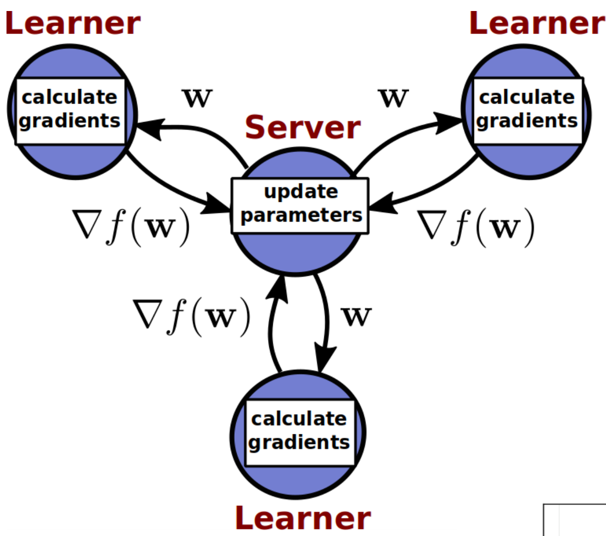


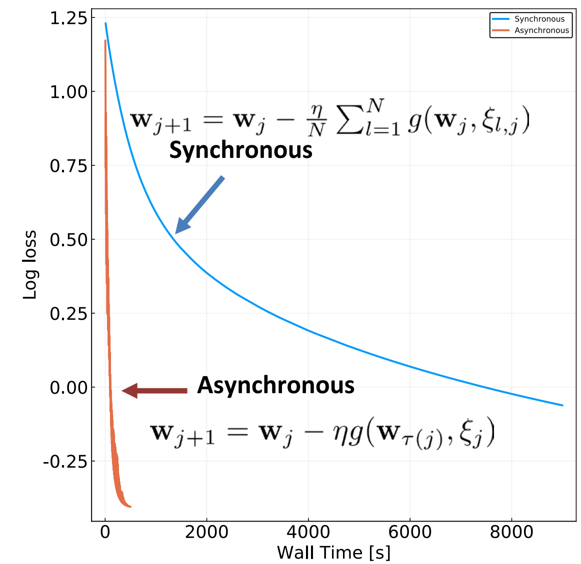
Simulating MPC Architectures

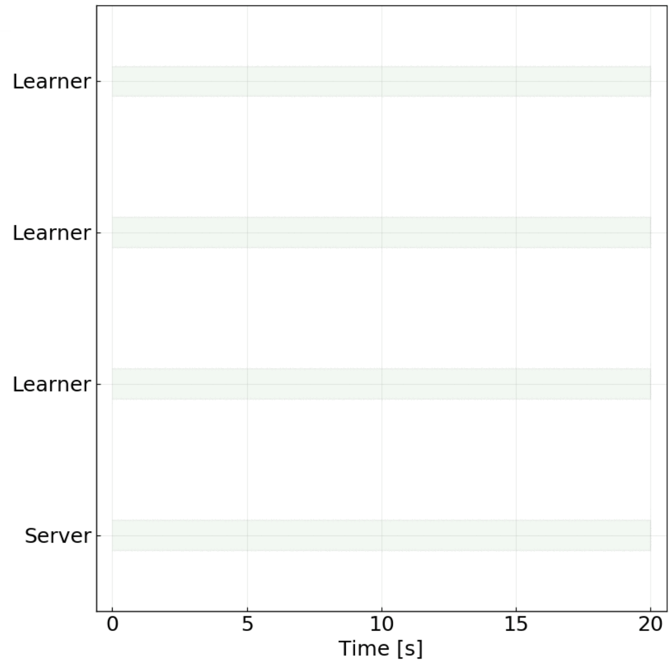


Key: Can simulate communication/computation latency & failures as well as energy usage.

Simulating Distributed Optimization Algorithms

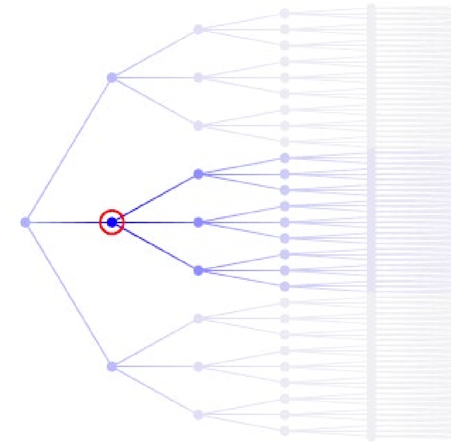




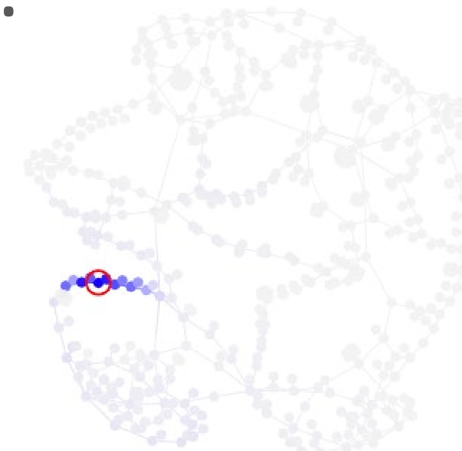


Key: Can simulate behavior of algorithms on specific hardware.

Argonne NATIONAL LABORATORY



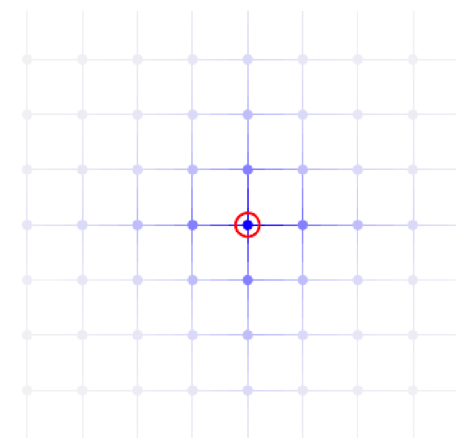
A Graph-Theoretic Perspective on Optimization:
Theory, Modeling, Algorithms, and Applications

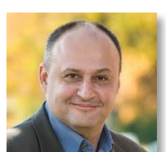


Victor M. Zavala

Department of Chemical & Biological Engineering
University of Wisconsin-Madison

Mathematics and Computer Science Division
Argonne National Laboratory



Work with: Sungho Shin, Mihai Anitescu, Jordan Jalving, David Cole

References

Jalving, J., Cao, Y., & Zavala, V. M. (2019). **Graph-based modeling and simulation of complex systems.** *Computers & Chemical Engineering*, *125*, 134-154.

Shin, S., Anitescu, M., & Zavala, V. M. (2022). **Exponential decay of sensitivity in graph-structured nonlinear programs. SIAM Journal on Optimization**, 32(2), 1156-1183.

Shin, S., & Zavala, V. M. (2021). **Controllability and observability imply exponential decay of sensitivity in dynamic optimization**. *IFAC-PapersOnLine*, *54*(6), 179-184.

Shin, S., Zavala, V. M., & Anitescu, M. (2020). **Decentralized schemes with overlap for solving graph-structured optimization problems**. *IEEE Transactions on Control of Network Systems*, *7*(3), 1225-1236.

Shin, S., Coffrin, C., Sundar, K., & Zavala, V. M. (2021). **Graph-based modeling and decomposition of energy infrastructures**. *IFAC-PapersOnLine*, *54*(3), 693-698.

Grüne, L., Schaller, M., & Schiela, A. (2020). **Exponential sensitivity and turnpike analysis for linear quadratic optimal control of general evolution equations.** *Journal of Differential Equations*, *268*(12), 7311-7341.

Shin, S., & Zavala, V. M. (2022). **Diffusing-horizon model predictive control.** *IEEE Transactions on Automatic Control*. In Press.

Shin, S., Faulwasser, T., Zanon, M., & Zavala, V. M. (2019). **A parallel decomposition scheme for solving long-horizon optimal control problems.** In *2019 IEEE 58th Conference on Decision and Control (CDC)* (pp. 5264-5271). IEEE.

https://github.com/zavalab/Plasmo.jl

https://github.com/MadNLP/MadNLP.jl