SAND2021-7885PE

Sandia National Laboratories

Exceptional service in the national interest

Exploring Optimization and Robust Control Techniques for use in Control Design of Nonlinear Dynamic Systems

Work Funded under IA A4H

Presenter: Justin Wilbanks

LDRD PI: David Kozlowski

LDRD Team Members: Kyle Williams, Rachel Schlossman, Gerard Bennett

2021 Engineering Sciences Research Foundation External Review

Flash Talks: Spanning R-D-A Spectrum in Engineering Sciences

July 13, 2021

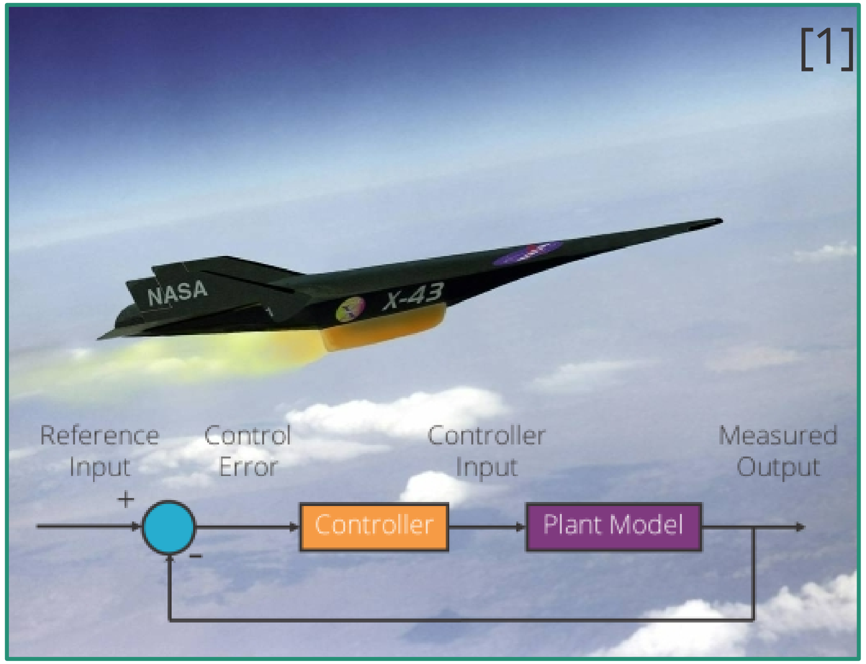
SAND 2021-XXXX P

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

U.S. DEPARTMENT OF ENERGY

NNSA

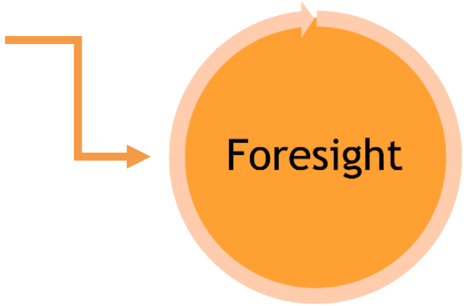
Background and Motivation



- Developing complex control systems can require substantial intuition and experience

- Traditional control design techniques can be combined with black-box or white-box optimization methods

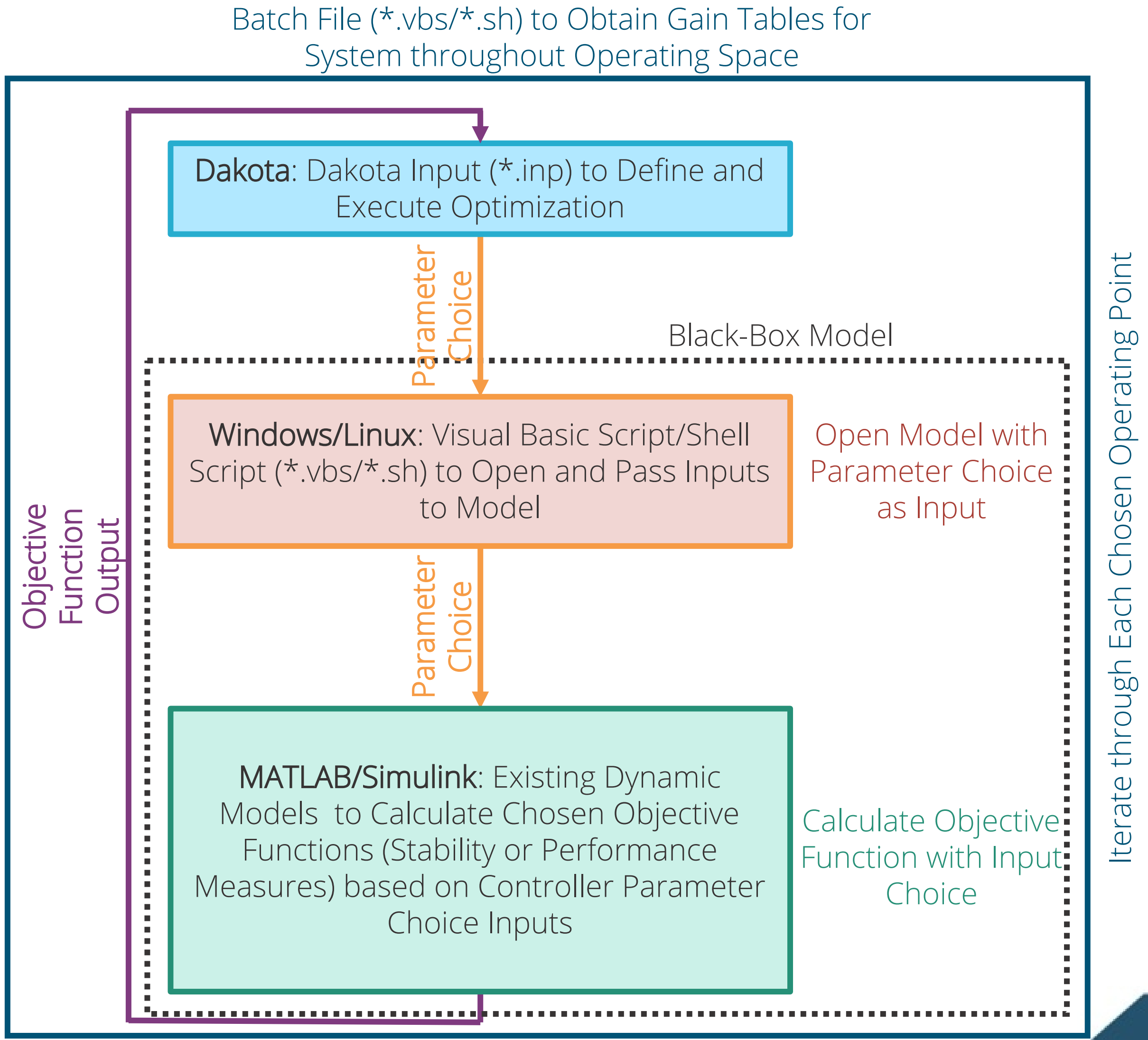
- Creating these tools now will increase the efficacy of the control design process and decrease the design burden



[1] Gibbs, Y. (2017). NASA Armstrong Fact Sheet: Hyper-X Program. NASA, Retrieved April 15, 2020 from https://www.nasa.gov/centers/armstrong/news/FactSheets/FS-040-DFRC.html.

Introduction of Dakota and Pyomo

- Dakota and Pyomo offer open-source solutions to optimization for control design [2,3]
  - Both were developed at Sandia National Laboratories
- They can be leveraged alongside models of dynamic systems and their controllers
  - Dakota can be applied to existing MATLAB/Simulink models
  - Pyomo requires a new model to be developed inside the Python-based environment
- Both approaches can be applied to complex nonlinear control problems



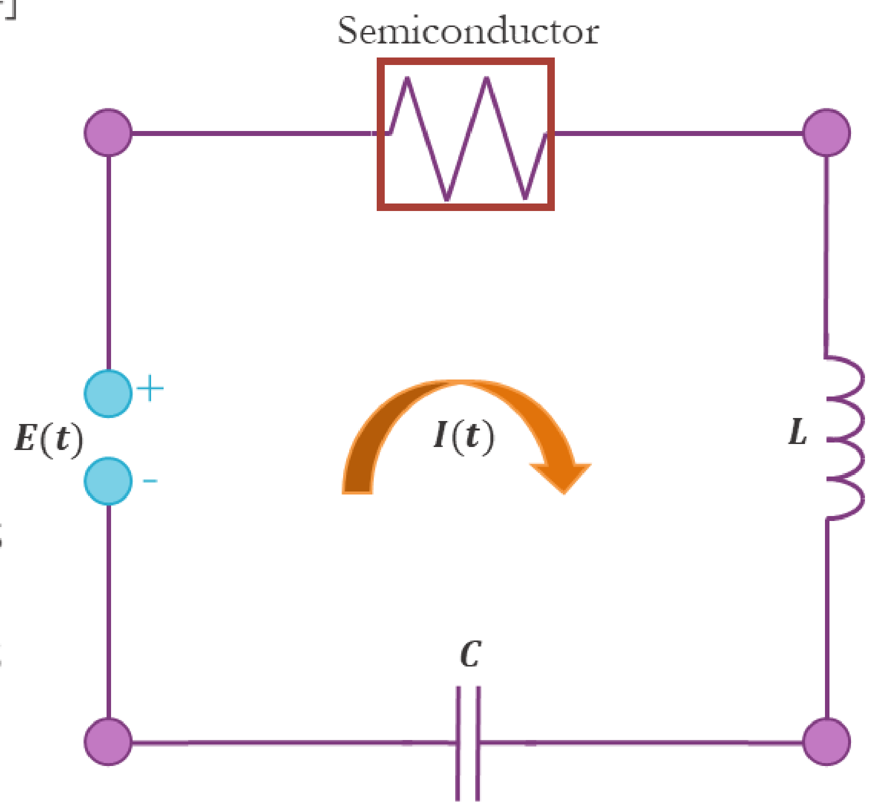
[2] Hart, W.E., Laird, C.D., Watson, J.P.,]Woodruff, D.L., Hackebeil, G.A., Nicholson, B.L., and Siirola, J.D. (2017). Pyomo – Optimization Modeling in Python. Second Edition. Vol. 67. Springer.
[3] Adams, B.M., Bohnhoff, W.J., Dalbey, K.R., Ebeida, M.S., Eddy, J.P., Eldred, M.S., Hooper, R.W., Hough, P.D., Hu, K.T., Jakeman, J.D., Khalil, M., Maupin, K.A., Monschke, J.A., Ridgway, E.M., Rushdi, A.A., Seidl, D.T., Stephens, J.A., Swiler, L.P., and Winokur, J.G. (2020). Dakota, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 6.12 User's Manual, Sandia Technical Report SAND2020-12495.

Example Nonlinear Dynamic System

- An example system to show the application of these methods is a Van der Pol (VDP) oscillator for optimizing feedback linearization (FBL) pole location [4]
- Using the circuit diagram representation, the equation of motion for the system is expressed as:

$$L\ddot{I} + \dot{I}(3bI^2 - a) + \frac{1}{C}I = E \qquad (1)$$

  - I: Current [A]
  - C: Capacitance [F]
  - L: Inductance [H]
  - a: Semiconductor parameter used in definition of voltage drop across element [Vs/A]
  - b: Semiconductor parameter used in definition of voltage drop across element [Vs/A^3]
  - E: Input voltage of the circuit [V]



[4] Borrelli, R.L. (1992). Differential Equations Laboratory Workbook: A Collection of Experiments, Explorations, and Modeling Projects for the Computer. John Wiley & Sons, Inc..

Control Applied to Example Nonlinear Dynamic System

- Expressing the VDP oscillator in state-space form:

$$x_1 = I \quad (2)$$

$$x_2 = \dot{I} \quad (3)$$

$$\dot{x}_1 = x_2 \quad (4)$$

$$\dot{x}_2 = \frac{1}{L}\left(-x_2(3bx_1^2 - a) - \frac{1}{C}x_1 + E\right) \quad (5)$$

- This leads to the following definition for the FBL controller input:

$$E = x_2(3bx_1^2 - a) + \frac{1}{C}x_1 + Lv \quad (6)$$

  - Using this input results in the following dynamics:

$$\dot{x}_1 = x_2 \quad (7)$$

$$\dot{x}_2 = v \quad (8)$$

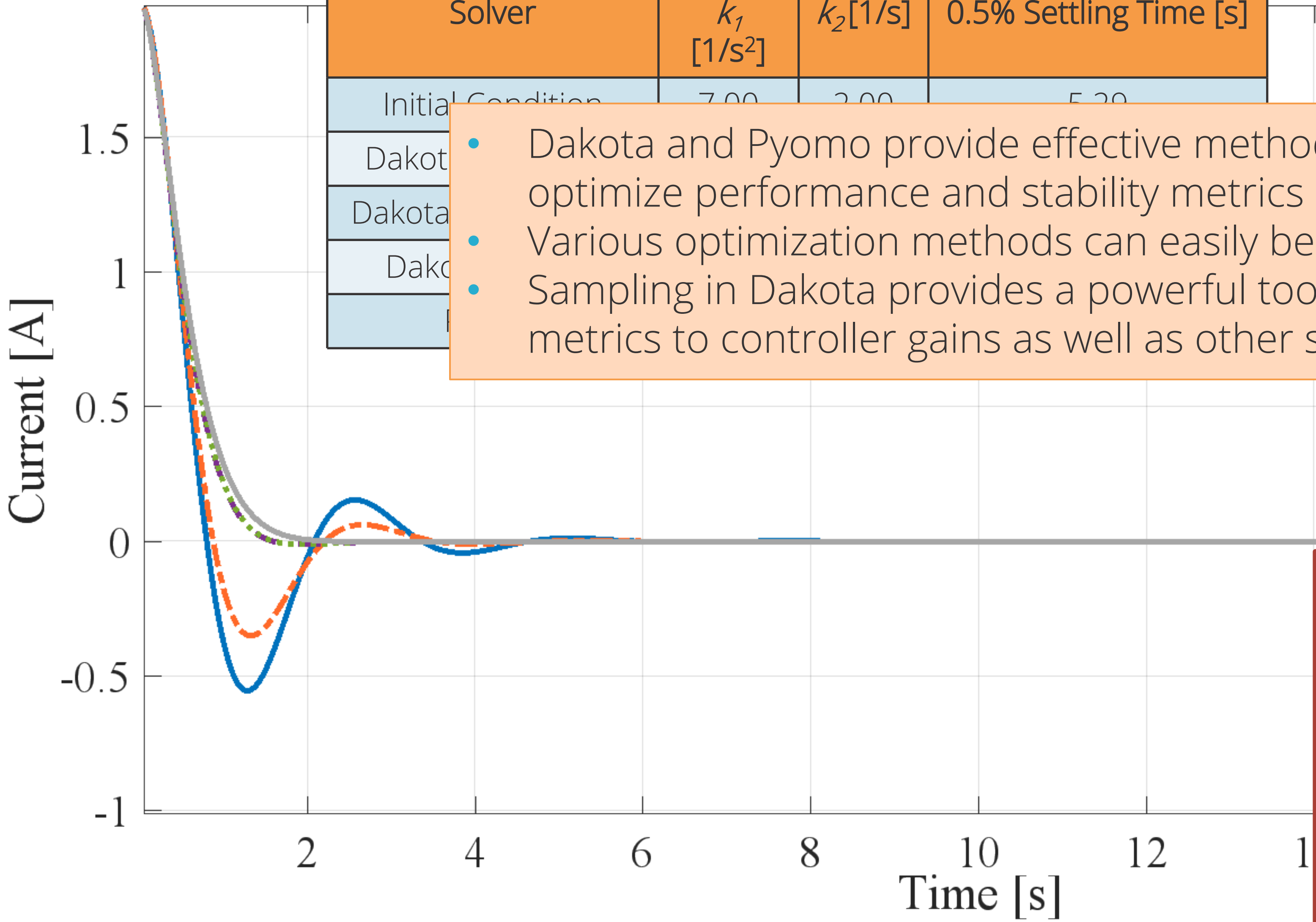
  - To allow for simple definition of the poles of the system, v can be selected as:

$$v = -k_1 x_1 - k_2 x_2 \quad (9)$$

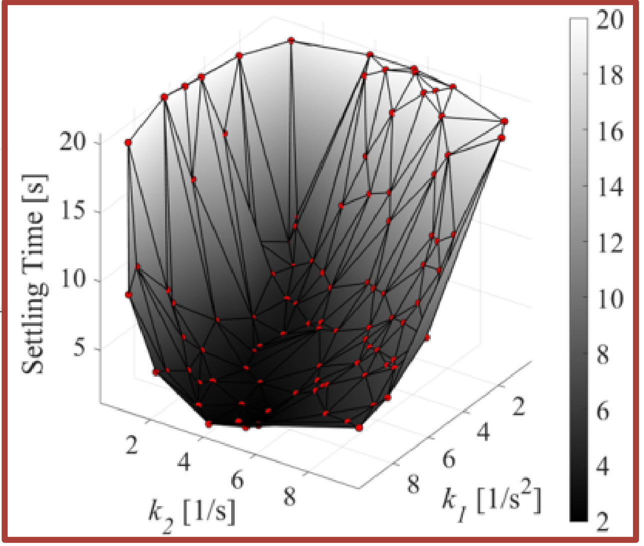
    - k_1 and k_2 are controller gains that can be selected for desirable system performance

Application of Dakota and Pyomo to Example System

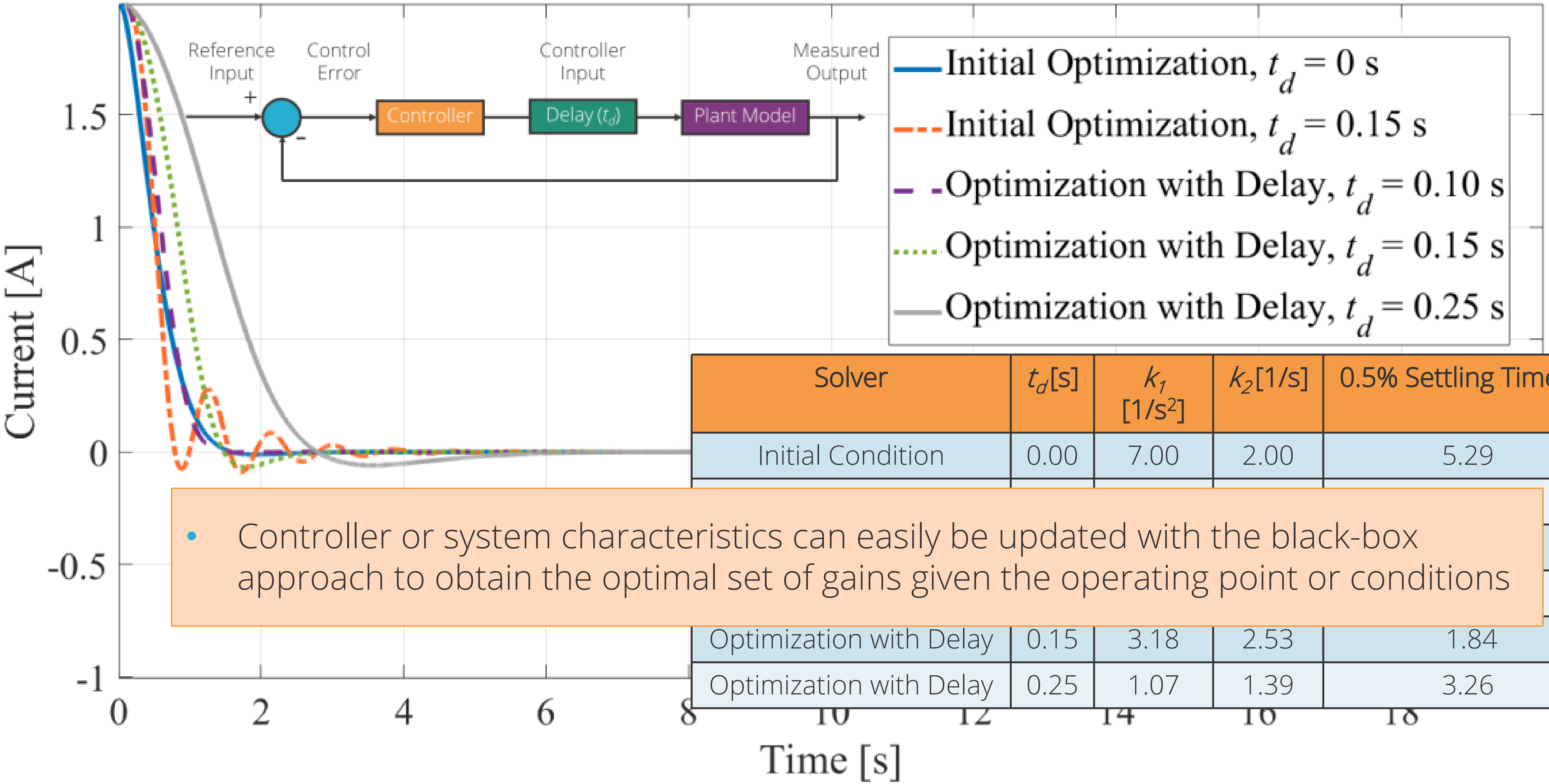
Solver	k_1 [1/s^2]	k_2[1/s]	0.5% Settling Time [s]
Initial Condition	7.00	2.00	5.29
Dakot			
Dakota			
Dako			
P			



- Dakota and Pyomo provide effective methods to choose controller gains to optimize performance and stability metrics
- Various optimization methods can easily be applied to improve results
- Sampling in Dakota provides a powerful tool to look at the sensitivity of metrics to controller gains as well as other system uncertainties



Developing Gains for Control Delay in Example System



Solver	t_d[s]	k_1 [1/s^2]	k_2[1/s]	0.5% Settling Time [s]
Initial Condition	0.00	7.00	2.00	5.29
Optimization with Delay	0.15	3.18	2.53	1.84
Optimization with Delay	0.25	1.07	1.39	3.26

- Controller or system characteristics can easily be updated with the black-box approach to obtain the optimal set of gains given the operating point or conditions

Advantages and Disadvantages of Dakota and Pyomo

- Mutual Advantages:
  - Both methods offer a powerful approach to determine effective control parameters throughout a system's operating space
  - Offers open-source solution to control development that decreases burden for new systems or staff

DAKOTA

- Advantages:
  - Can leverage existing models in tools like MATLAB
  - Built-in capabilities of MATLAB/Simulink can be leveraged in the black-box approach
- Disadvantages:
  - Computation time is directly related to simulation time of model

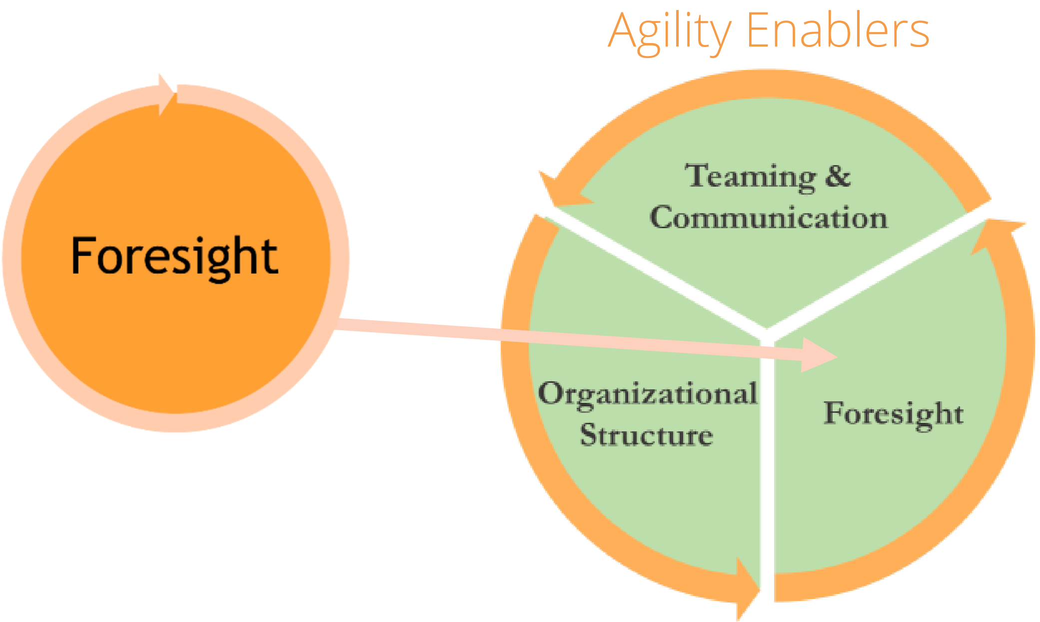
PYOMO

- Advantages:
  - Computation times are significantly faster (~1/10th of equivalent Dakota run)
- Disadvantages:
  - Significant model development can be onerous and time consuming
  - Some control-related metrics are difficult to calculate within existing Python-based Pyomo framework

Conclusions

- Dakota and Pyomo offer powerful open-source solutions to decrease the development burden of new controllers and dynamic systems
  - Beginning to introduce these tools in the design process enables agility in control design and system development
  - These tools also allow staff members to learn more about the influence of system/controller parameters on system response

- These tools and workflows provide the foresight to address the future impact of rapidly changing operating spaces or reduced design cycles on the control design and dynamic modeling process



Acknowledgements

- Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525

- Thank you for your attention! Are there any questions?

References

[1] Gibbs, Y. (2017). NASA Armstrong Fact Sheet: Hyper-X Program. NASA, Retrieved April 15, 2020 from https://www.nasa.gov/centers/armstrong/news/FactSheets/FS-040-DFRC.html.

[2] Hart, W.E., Laird, C.D., Watson, J.P.,]Woodruff, D.L., Hackebeil, G.A., Nicholson, B.L., and Siirola, J.D. (2017). Pyomo – Optimization Modeling in Python. Second Edition. Vol. 67. Springer.

[3] Adams, B.M., Bohnhoff, W.J., Dalbey, K.R., Ebeida, M.S., Eddy, J.P., Eldred, M.S., Hooper, R.W., Hough, P.D., Hu, K.T., Jakeman, J.D., Khalil, M., Maupin, K.A., Monschke, J.A., Ridgway, E.M., Rushdi, A.A., Seidl, D.T., Stephens, J.A., Swiler, L.P., and Winokur, J.G. (2020). Dakota, A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 6.12 User's Manual, Sandia Technical Report SAND2020-12495.

[4] Borrelli, R.L. (1992). Differential Equations Laboratory Workbook: A Collection of Experiments, Explorations, and Modeling Projects for the Computer. John Wiley & Sons, Inc..