# MECH 6v29.002 – Model Predictive Control

Tuesday and Thursday 8:30 – 9:45am

L1 - Intro

# Introduction

- Dr. Justin Koeln
  - Office: ECSW 3.355D
  - Office Hours: Tuesday, Thursday 10-11am (after class)
  - Email: Justin.Koeln@UTDallas.edu
    - I can answer quick questions via email

# Who am I?

- Assistant Professor in Mechanical Engineering

**Logan, UT**
**2007-2011**
B.S. in Mechanical Engineering

**Champaign, IL**
**2011-2017**
MS, PhD, Postdoc in Mechanical Engineering

**Germantown, MD**

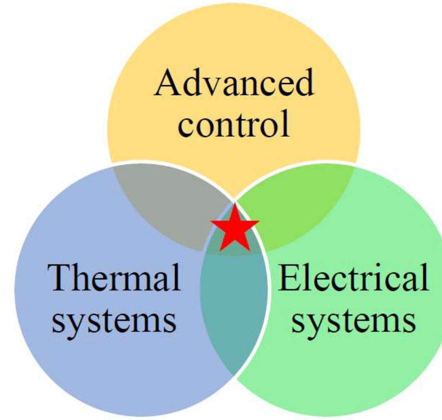
**Dallas, TX**
**2018-Present**
Assistant Professor, Mechanical Engineering

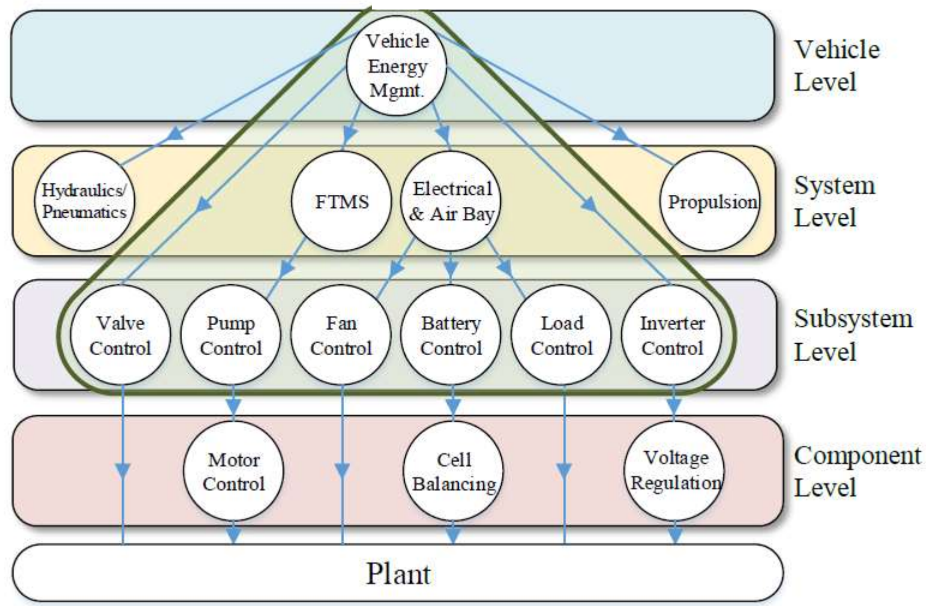
# Who am I?

- UTD

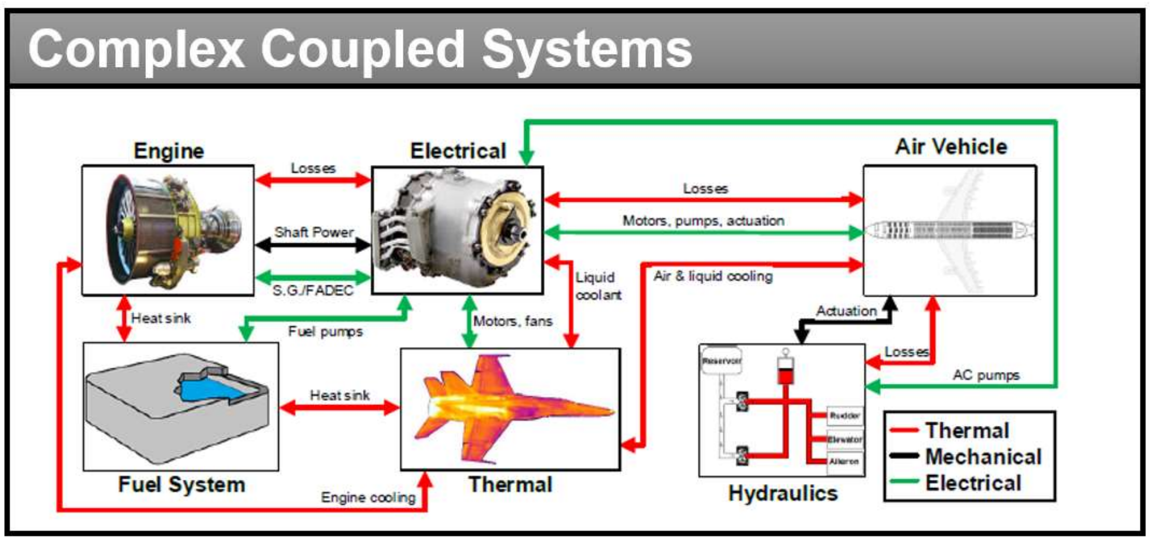
UTD Energy Systems Control Laboratory

- Research Focus
  - Dynamic modeling and control of complex energy systems




Distributed and Hierarchical Control Strategies



# Who am I?

- Scalable and computationally efficient methods for Model Predictive Control
  - ~10 out of 23 journal papers
  - ~15 out of 26 conference papers
  - Theoretical and practical
  - Simulation and experiment
  - Hierarchical MPC
  - Nonlinear MPC
  - Robust MPC (for guaranteed constraint satisfaction)
  - Set-based methods

# Who are you?

- Please introduce yourself
  - Name (say it slowly)
  - MS or PhD program
  - # of years at UTD
  - Research interests (if applicable)
  - Research advisor (if applicable)
  - Something you hope will be interesting about this course
  - Something you hope to take away from this course
  - Anything else you want to share

# Syllabus and Course Expectations

- Any questions about the syllabus?
- What is an important course expectation that we should agree on?

# Course Objectives

- Intro to Model Predictive Control (MPC)
  - Leading approach to optimizing control of multivariable system subject to constrained operation
  - We will focus on:
    - Theory: learning the theory and analysis techniques used to guarantee closed-loop control properties like stability and constraint satisfaction
    - Application: the practical aspects of designing and implementing MPC for a variety of engineering systems
- By the end of this course, you should be able to:
  - Formulate common MPC optimization problems and analyze the theoretical properties of the closed-loop system
  - Design, tune, and implement MPC controllers in simulation
  - Summarize the key assumptions and analysis techniques of novel MPC formulations from the literature and showcase the features of these formulations with simulation results

# Course Website

UTD eLearning: https://elearning.utdallas.edu/

- Includes course administrative and schedule information
- Used to post homework, project description, grades, announcements, etc.
- Check it frequently

- No textbook
  - Lectures will primary source of information
  - Research papers and textbooks (available for free online) will be used for supplemental reading

No information (assignments, recorded lectures, etc.) from this course should be shared/posted outside this class.

# Course Assumptions/Expectations

- No official prerequisite
- Will provide a brief review of math preliminaries
- I will assume a strong understanding of Linear Systems
  - MECH 6300 (or equivalent)
    - Dynamic systems representations (state-space)
    - Linear systems analysis (eigenvalues, similarity transforms, etc.)
    - Lyapunov stability
    - Controllability/observability
    - LQR
- Attend live lectures and participate (ask/answer questions)
- Keep pace with the course, HW deadlines, etc.
- Can work with each other but no sharing of solutions or code.
- Email me with any questions/concerns

# Software

- In class examples and homework assignments for this course will be completed using Matlab/Simulink
- You are expected to have consistent/frequent access to a computer with Matlab/Simulink
- Some assignments might require the use of addition toolboxes/extensions for Matlab such as
  - Multi-Parametric Toolbox (MPT) - https://www.mpt3.org/
  - YALMIP (included with MPT) - https://yalmip.github.io/
  - GUROBI - https://www.gurobi.com/
- While there are many other software packages that perform similar functions, these tools will be the ones used by the instructor throughout the class and it is expected that you use these same tools

# Lectures, Assignments, and Grading

- Lectures will be twice a week Tuesday and Thursday 8:30 – 9:45am
- Some lectures will have a short exercise to be completed after the lecture
  - Paper reading, Short quiz on eLearning, Small Matlab example/demonstration
  - These will supplement the homework and are designed to be relatively easy and help prepare you for the homework
- There will be 4 homework assignments throughout the semester (roughly every other week)
- Middle to end of semester will focus on individual project
- Grading
  - Participation: 20%
  - Homework: 40%
  - Project Presentation: 20%
  - Project Report: 20%

# Homework and Project

- Homework is to be submitted via eLearning by 5pm on Fridays based on schedule
- Submitted as a single PDF
- Answers must be neat, organized, and typed
  - Good practice for writing research papers
- While you are encouraged to learn with your fellow students, simply copying MATLAB/Simulink code/files or written reports from others is cheating

Project:

- More details to come
- Opportunity for independent study of an advanced MPC technique of your choosing
- Outcome will be a conference-type paper/report and presentation

# Outline/Schedule

Subject to change

| Week of: | Topic | Due |
|---|---|---|
| 8/21 | Introduction to MPC – Key Concepts | |
| 8/28 | Mathematical Background and Dynamic Systems | |
| 9/04 | (Monday – No Lecture - Labor Day) MPC Theory – Stability | |
| 9/11 | Unconstrained MPC and Extensions | HW #1 |
| 9/18 | MPC Theory – Feasibility | |
| 9/25 | MPC Theory – Invariant Sets and Persistent Feasibility | HW #2 |
| 10/02 | Robust MPC | |
| 10/09 | Robust MPC | Project Proposal |
| 10/16 | MPC Development and Applications | HW #3 |
| 10/23 | Project Discussions | |
| 10/30 | Nonlinear MPC | |
| 11/06 | Decentralized and Distributed MPC | HW #4 |
| 11/13 | Explicit and Hybrid MPC | |
| 11/20 | No Lectures (Fall Break) | |
| 11/27 | Project Presentations | |
| 12/04 | No Lectures (Last Week of classes) | Project Report |

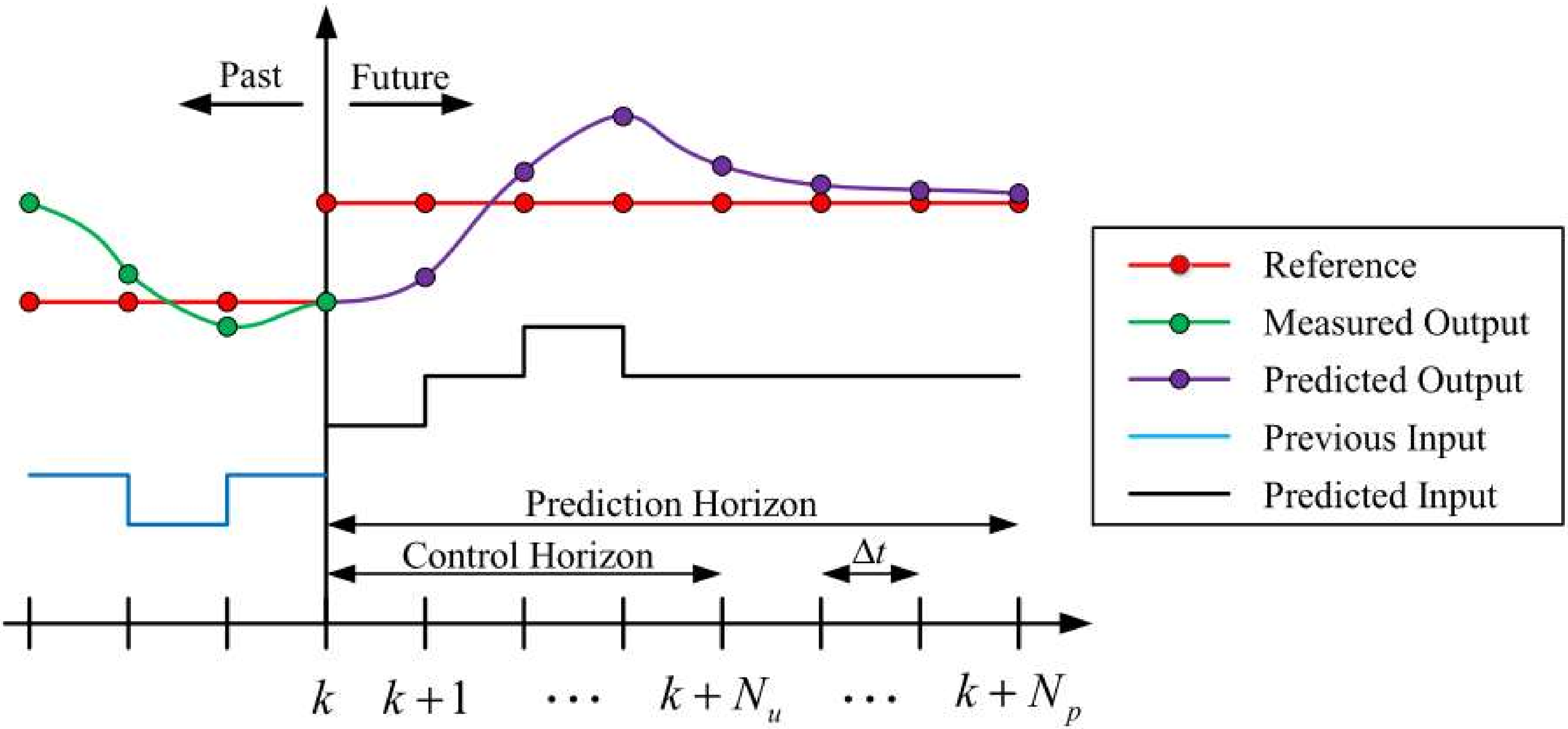
# Any Questions?

# Intro to Model Predictive Control

- What is MPC?
  - A receding-horizon optimal control framework that uses a dynamic model of a system to predict the future response of the system
  - By solving a finite-time horizon, open-loop, optimal control problem using the current state of the system, MPC determines a sequence of control decisions which minimize the specified cost function over the prediction horizon
  - The first element of this control sequence is applied to the system and the procedure is repeated at the next time instance.

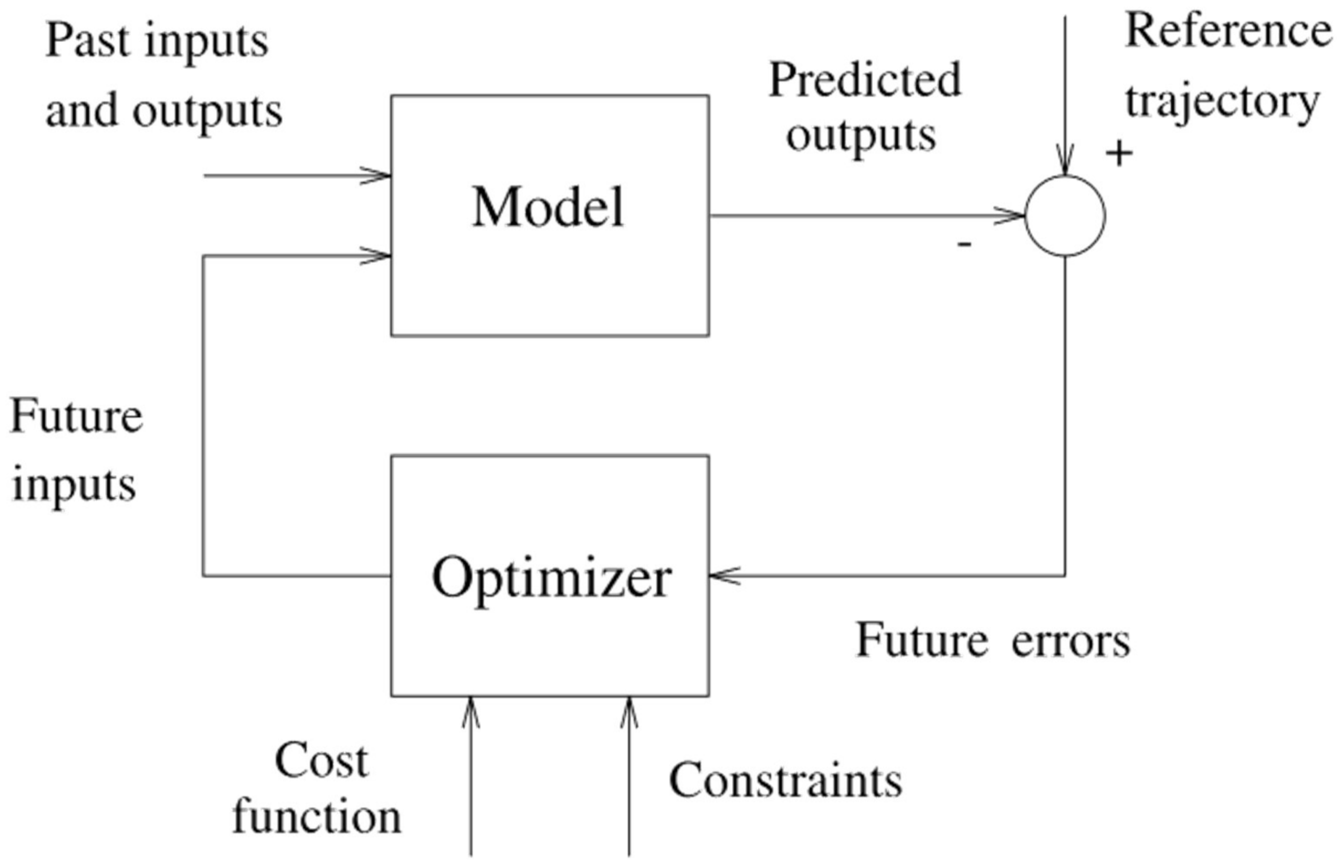
# Intro to Model Predictive Control

- What is MPC?



# Intro to Model Predictive Control

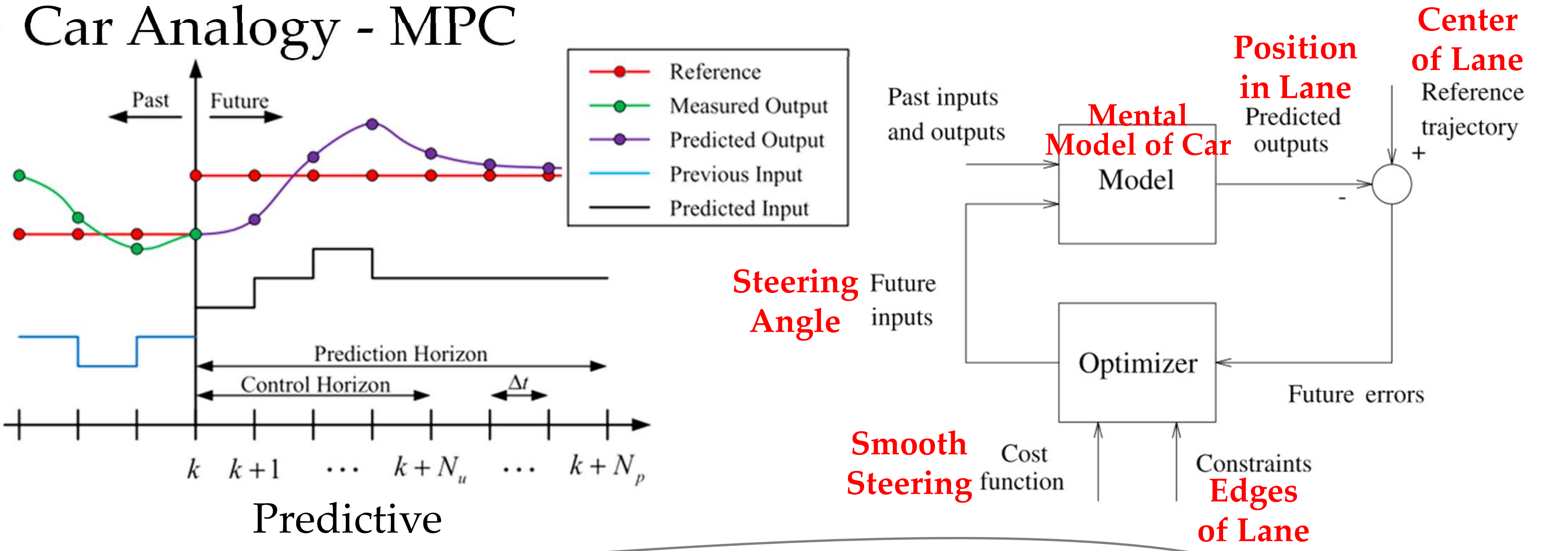
- What is MPC?

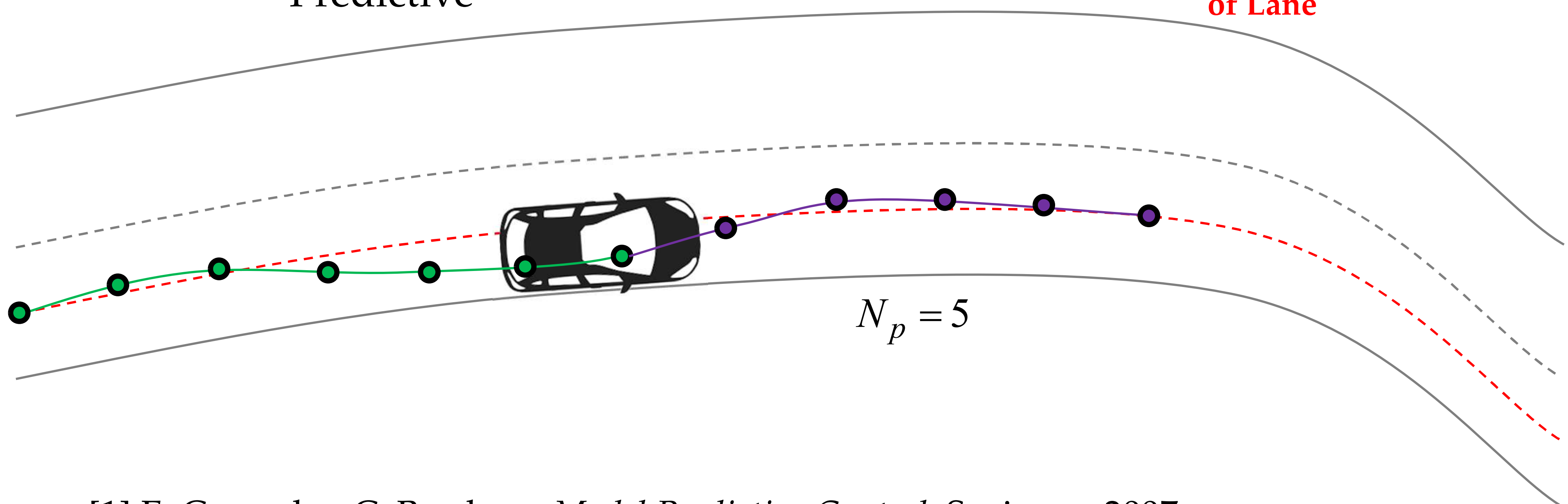



[1] E. Camacho, C. Bordons. *Model Predictive Control*. Springer, 2007.

# Intro to Model Predictive Control

- Car Analogy - MPC





[1] E. Camacho, C. Bordons. *Model Predictive Control*. Springer, 2007.

# Intro to Model Predictive Control

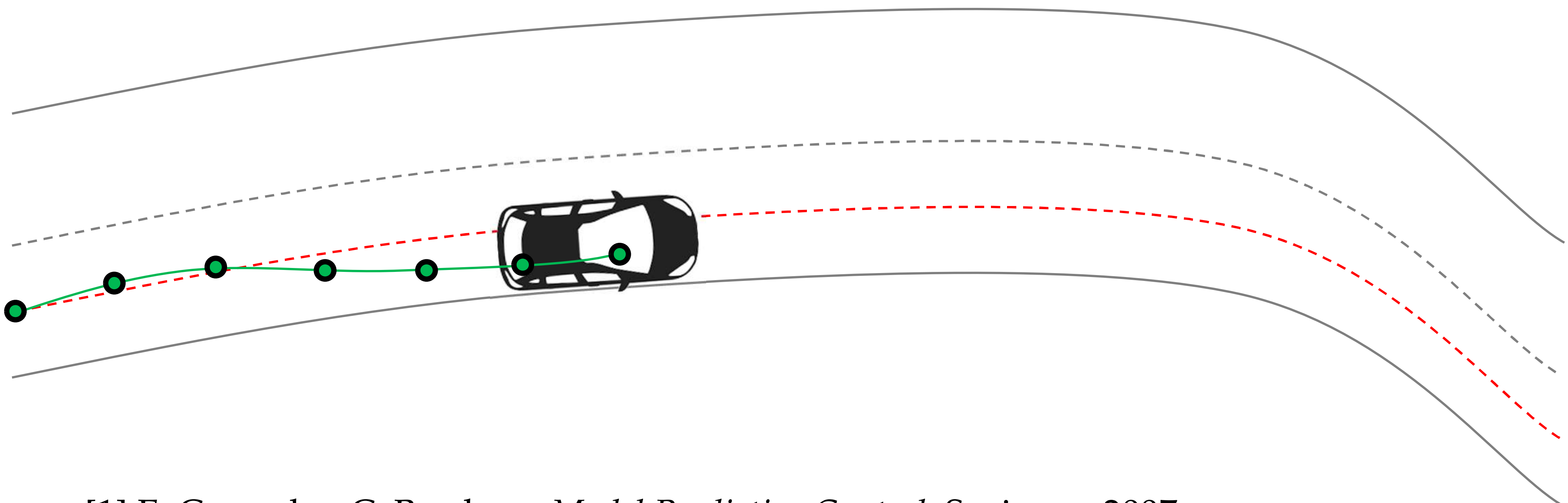
- Car Analogy - PID

$$u = k_p e + k_i \int e\, dt + k_d \dot{e}$$

**Steering is only based on past tracking error**

↓

**Drive only using side/rear view mirrors**



[1] E. Camacho, C. Bordons. *Model Predictive Control*. Springer, 2007.

# Intro to Model Predictive Control

- Mathematical Formulation

Prediction horizon length: $N$
Input sequence: $U_t = \{u_t, u_{t+1}, \ldots, u_{t+N-1}\}$

Cost Function to be minimized via optimization

$$\min_{U_t} \sum_{k=t}^{t+N-1} J(x_k, u_k) + J_T(x_{t+N}, u_{t+N})$$

subject to:

$$s.t.\ \forall k = t, \ldots, t+N-1$$

- Dynamic system constraints (typically state-space model)

$$x_{k+1} = f(x_k, u_k),$$

- Input, state, and output constraints

$$u_k \in \mathcal{U},\ x_k \in \mathcal{X},\ y_k = g(x_k, u_k) \in \mathcal{Y}$$

- Initial condition constraint

$$x_t = x(t)$$

1. Measure current state: $x(t)$
2. Solve for optimal input sequence: $U_t^*$
3. Only apply first input: $u_t^*$
4. Repeat at $t$+1

# Intro to Model Predictive Control

- Key issues

  Planned open-loop trajectories $\neq$ closed-loop trajectories

  - even if you have a perfect model and no disturbances
  - often due to finite prediction horizon
- Feasibility
  - How can we guarantee that the optimization problem will always have at least one solution that satisfies all constraints?
- Stability
  - How can we guarantee the closed-loop system is stable?
  - Not as easy as checking eigenvalues since we are solving an optimization problem at each time-step.
- Constraint satisfaction
  - How can we guarantee that the closed-loop state and input trajectories satisfy the desired constraints?
- Real-time implementation
  - Will we be able to solve the optimization problem fast enough?