MEEN 432 –Automotive Engineering

Fall 2026

Instructor: Dr. Arnold Muyshondt

Acknowledgement: Most of the material for this class was developed by Dr. Swami Gopalswamy

Lecture 3: Components Based Modeling

- Components Based Modeling

Modeling

- Modeling is a powerful way to understand a physical system's behavior / performance
- In turn, this enables:
  - Development of optimal designs
  - Confirmation of adequate performance of your system over a large variety of scenarios
- Models are "abstractions" of your real system
  - Designed to answer specific questions about your system
  - Different methods exist for modeling, depending on the modeling goal

Modeling Methods

- Finite Element Modeling and Analysis
  - Detailed Structural Performance, Computational Fluid Dynamics, …
- Excel Spreadsheet Analysis
  - Thumb rules useful for design captured in spreadsheets
    - Relying on engineering intuition and experience
  - Mostly used for “steady state” analysis, though complex dynamic analysis can be developed
  - Very popular user-interface
- Dynamic Systems Modeling (relevant to our course)
  - ODE representation of dynamics
    - “Feedback Control Systems” → Model Based Control development
      - Model Based Development, Model Based Design, Model Based Systems Engineering ,…

Requirements for Good Modeling

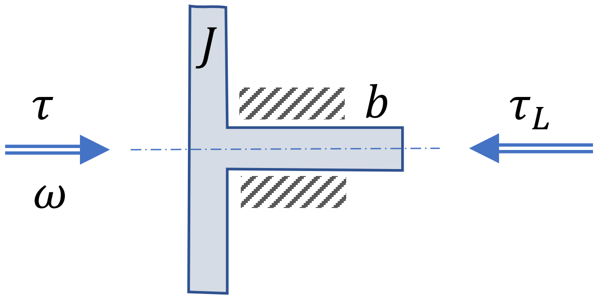
- Models are critical "assets" – they capture the knowledge base in a company, even more than real assets (i.e. prototypes)
  - Highly Complex
  - Developed by many engineers (10's to 100's)
  - Used by many many engineers (100's to 1000's)
- Principles to support good models
  - Modularity
  - Re-usability
  - Readability
  - Maintainability
  - Testability

Modeling to reflect a "parts-based decomposition" of the System:

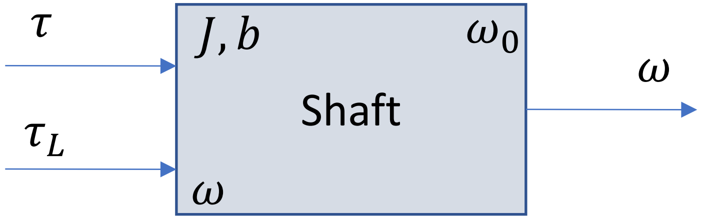
- Model the parts (sub-systems) separately, and "put them together" to realize the full vehicle (system)

Parts Based Decomposition Modeling

- Key requirement for parts-based decomposition is understanding the interface between the parts
  - What variables are determined (calculated) where, and what variables are inputs where, and outputs where.
- Consider the simple mechanical shaft:
  - $J\,\dot{\omega} = \tau - \tau_L - b\,\omega$

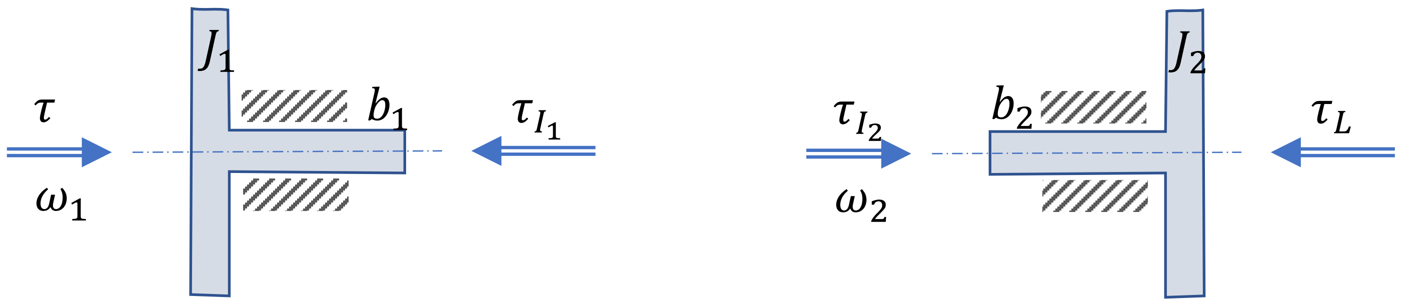


- τ, τ_L are inputs to the component
- ω is a state, and an output of the component
- J, b are parameters of the component



System Integration

- How can you combine two parts?
  - Example: Two simple shafts "connected" together
- Two considerations:
  - Kinematic constraint: $\omega_1 \equiv \omega_2 \rightarrow \dot{\omega}_1 = \dot{\omega}_2, \ddot{\omega}_1 = \ddot{\omega}_2, \ldots$
  - Physical Connection constraint: $\tau_{I_1} = \tau_{I_2}$ (Newton's Action-Reaction Law)
- How can we impose these constraints?
  - while still maintaining modularity of the models?

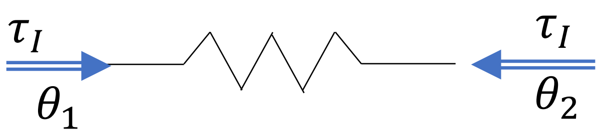


System Integration

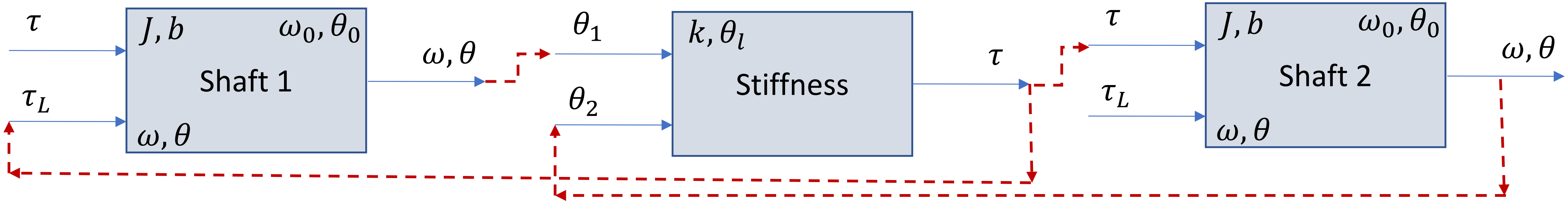
- Option 0 (Symbolic Solution):
  - Generate new equations based on the constraint.
  - $\omega_1 = \omega_2 \coloneqq \omega \, ; \; \dot{\omega}_1 = \dot{\omega}_2 \coloneqq \dot{\omega} \Rightarrow \frac{1}{J_1}\left(\tau - \tau_{I_1} - b_1\omega\right) = \frac{1}{J_2}(\tau_{I_2} - \tau_L - b_2\omega)$
  - $\tau_{I_1} = \tau_{I_2} \coloneqq \tau_I \Rightarrow \tau_I = \frac{1}{J_1+J_2}(J_2\tau + J_1\tau_L) - \frac{1}{J_1+J_2}(b_1J_2 - b_2J_1)\omega$
  - Substituting back:
    - $J_1\dot{\omega} = (\tau - \tau_I - b_1\omega) = \frac{J_1}{J_1+J_2}(\tau - \tau_L)\,) + \frac{1}{J_1+J_2}(b_1J_2 - b_2J_1 - b_1J_1 - b_1J_2)\omega$
  - Or, $\dot{\omega} = \frac{1}{J_1+J_2}(\tau - \tau_L) - \frac{b_1+b_2}{J_1+J_2}\omega \coloneqq \frac{1}{J_c}(\tau - \tau_L) - \frac{b_c}{J_c}\omega$
- This can also be obtained intuitively by observing that when we combine we are simply getting a larger inertia and damping!
- However, by doing this, we lose the ability to model the two shafts separately
  - That is a challenge we want to carefully address
- Real Life engineering is done by teams of engineers distributed across space and time!

System Integration

$$\tau_I = k\,(\theta_2 - \theta_1 - \theta_l)$$



- Option I (Stiff Connection):
  - You can model the connection between the two shafts as a “very high stiffness spring”
  - Constraint:
    - We only have the physical connection constraint: $\tau_{I_1} = \tau_{I_2} = \tau_I = k(\theta_1 - \theta_2 - \theta_l)$
    - There is no kinematic constraint – however we need to keep track of the angle θ (integral of ω) on each side of the spring stiffness
  - Challenge: High Stiffness → Slow simulation; Low Stiffness → Inaccurate approximation



System Integration

- Option 2: Share Inertia and Torque together
  - The idea is to decide the number of degrees of freedom of the overall system, and restrict the number of integrators in each component, so that the total number of degrees of freedom of the system is not violated
  - In our example, individual shaft d.o.f is 1. However, when combined it is still 1. Therefore, only one of the shaft models can have an integrator.
  - Upstream Component receives speed and position value from downstream component
    - In return, shares both its torque and inertia to downstream component
  - Receiving Component incorporates the torque and inertia of Upstream component
  - performs integration and shares calculated stats back to upstream component

    - $\dot{\omega} = \frac{\tau_{upstream} + \tau_{otherInputs} - \tau_{drag}}{J_{upstream} + J_{component}}$