Simulation Tools

Project 2

Simulation of Differential-Algebraic Equations

Campoli, Lucas

Cullen de Erquiaga, Magdalena Itziar

Ghigliani, Franco Agustín

Mutz, Matias

Exchange Students

LUND UNIVERSITY

LTH

FACULTY OF ENGINEERING

March 16, 2025

Contents

Introduction

This report presents Part II of the course project, focusing on the implementation and analysis of constrained ordinary differential equations (DAEs). Specifically, we investigate the effect of index reduction on the squeezer mechanism example described in the lecture.

Problem Description

The seven-bar mechanism consists of seven rigid bars connected by joints, forming a closed kinematic chain. The system is driven by a constant torque ($0.033\ \mathrm{N \cdot m}$) and includes a spring force (spring constant 4530 N/m, natural length 0.07785 m), making it a non-conservative system.

Key characteristics of the system:

- 7 rigid bars with specific masses (ranging from 0.00365 to 0.07050 kg) and moments of inertia
- Geometric constraints defining the connections between bars
- Constant driving torque of $0.033\ \mathrm{N \cdot m}$
- Spring force acting between specific points with c0 = 4530 N/m

Implementation

The implementation consists of two main Python files:

- `squeezer.py`: Contains the core implementation:
  - `res_general`: Central function handling all formulations through an index parameter
  - Class hierarchy for different formulations, each inheriting from the previous
  - Implementation of system dynamics, constraints, and initial conditions
- `main.py`: Handles simulation and visualization:
  - `run_seven_bar_problem`: Main function to run simulations with configurable parameters
  - `plot_soln`: Generates three types of plots (angles, velocities, multipliers)
  - `plot_stats`: Creates performance statistics visualizations

System Formulation

We implemented four different formulations of the problem using a class hierarchy in Python:

- **Index-3 formulation** (`Seven_bar_mechanism_indx3`): The original DAE system with position-level constraints
- **Index-2 formulation** (`Seven_bar_mechanism_indx2`): First derivative of constraints
- **Index-1 formulation** (`Seven_bar_mechanism_indx1`): Second derivative of constraints
- **Explicit formulation** (`Seven_bar_mechanism_expl`): Reformulated as an ODE system

All formulations share the same initial conditions:

- Initial angles: $\beta = -0.0617$, $\gamma = 0.4553$, $\phi = 0.2227$, etc.
- Initial velocities: All zero except $\ddot{\beta} = 14222.44$ and $\ddot{\Theta} = -10666.83$
- Initial Lagrange multipliers: $\lambda_1 = 98.57$, $\lambda_2 = -6.12$

The state variables include:

- 7 angles (β, Θ, γ, ϕ, δ, Ω, ϵ)
- 7 angular velocities ($\dot{\beta}$, $\dot{\Theta}$, $\dot{\gamma}$, $\dot{\phi}$, $\dot{\delta}$, $\dot{\Omega}$, $\dot{\epsilon}$)
- 6 Lagrange multipliers (λ_1 through λ_6)

Numerical Methods

We employed two main numerical solvers:

- IDA (Implicit Differential-Algebraic solver) for the DAE formulations
- Runge-Kutta 4th order method for the explicit formulation

The IDA solver was configured with:

- Absolute tolerances for velocities and Lagrange multipliers
- Algebraic variable flags for different components
- Option to suppress algebraic variables

Test Description and Methodology

To investigate the effects of index reduction and solver behavior, we conducted several systematic tests:

0.1 Test Cases

- **Index Comparison Test**: We compared all four formulations (Index-3, Index-2, Index-1, and Explicit) to analyze the effect of index reduction on solution accuracy and computational efficiency.
- **Solver Configuration Test**: For the Index-1 formulation, we tested different solver parameters:
  - Absolute tolerances: 1e-06 for positions, 1e+05 for velocities and multipliers
  - Algebraic variables suppression enabled
  - BDF method with maximum order 5
- **Time Step Analysis**: We examined the system's behavior over a 0.03-second interval, chosen to capture multiple oscillation periods of the mechanism.

0.2 Reproducibility

To reproduce these tests:

- Use Python with Assimulo 3.4
- Run the main script with default parameters: `python main.py`
- For index comparison, modify the `problem_index` parameter (0 for explicit, 1-3 for DAE formulations)
- Results (plots and statistics) are automatically saved in the `report/plots/` directory

Results and Analysis

We conducted several numerical experiments to analyze the system's behavior and compare different formulations. The results are presented in the following sections.

0.3 Solution Analysis

The simulation results show the time evolution of the system's state variables. Figure 1 displays the angular positions of all seven bars over time, revealing several interesting characteristics of the mechanism's behavior:

- The angle β shows a monotonic increase from 0 to approximately 15 radians over the simulation period, indicating continuous rotation of this bar
- The angle Θ exhibits the opposite behavior, decreasing monotonically from 0 to approximately -15 radians, suggesting a counter-rotating motion

- The remaining angles (γ, ϕ, δ, Ω, ϵ) oscillate within a much smaller range (approximately ± 1 radian), maintaining the mechanism's geometric constraints
- The symmetry in the oscillatory behavior of these angles indicates a well-balanced mechanical system

This behavior is consistent with what we would expect from a seven-bar mechanism where two bars drive the system while the others maintain the kinematic chain's integrity.

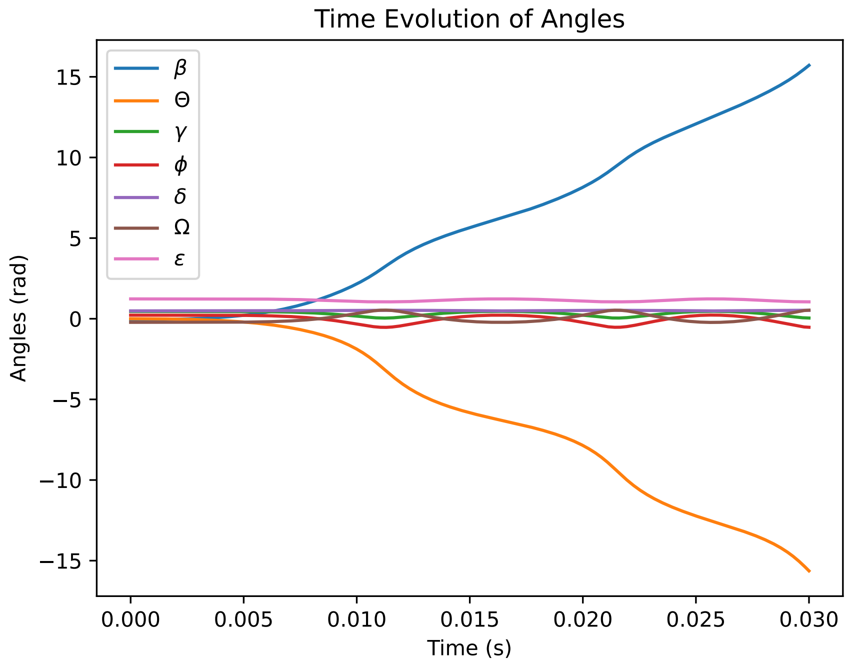

Figure 1: Time evolution of the seven angles in the mechanism

The angular velocities, shown in Figure 2, reveal the dynamic behavior of the mechanism. The analysis of these velocities shows several key features:

- $\dot{\beta}$ exhibits a non-linear increasing trend with periodic variations, reaching peaks of approximately 1000 rad/s, indicating an accelerating rotation with superimposed oscillations
- $\dot{\Theta}$ shows a similar but negative pattern, with velocities reaching -1500 rad/s, demonstrating the counter-rotation necessary to maintain the mechanism's motion
- The remaining angular velocities ($\dot{\gamma}$, $\dot{\phi}$, $\dot{\delta}$, $\dot{\Omega}$, $\dot{\epsilon}$) display periodic oscillations with smaller amplitudes (approximately ± 300 rad/s)
- The oscillation patterns are regular and phase-shifted relative to each other, indicating coordinated motion between the connecting bars
- Three distinct frequency components are visible in the motion:
  - A primary oscillation with a period of approximately 0.01 seconds
  - Higher frequency oscillations superimposed on the main motion
  - A gradual increase in the amplitude of oscillations over time

These velocity patterns are characteristic of a constrained mechanical system where the motion of each component is coupled through the geometric constraints, resulting in complex but coordinated dynamic behavior.

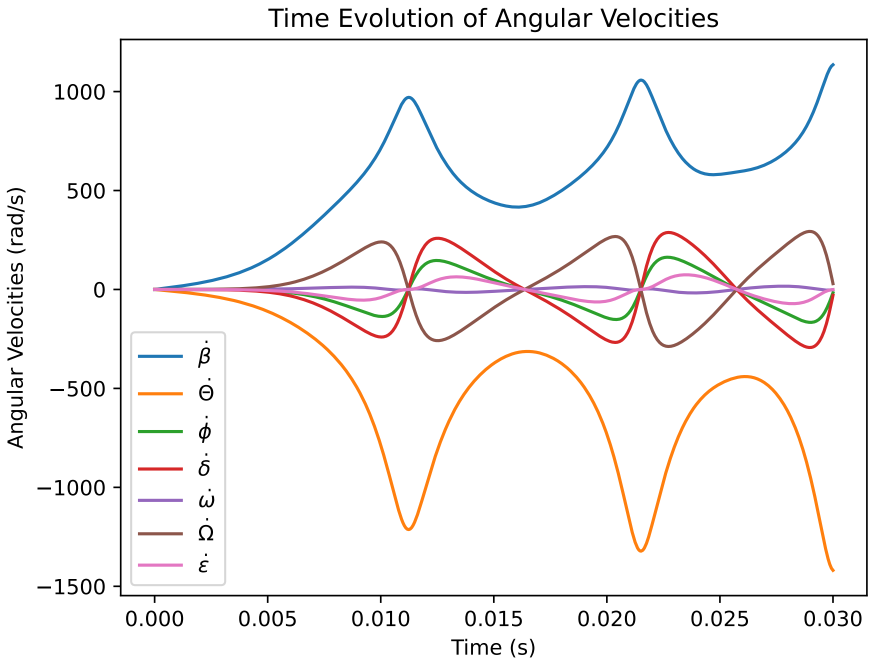

Figure 2: Time evolution of the angular velocities

The Lagrange multipliers, presented in Figure 3, represent the constraint forces in the system and provide crucial information about the internal dynamics of the mechanism. The analysis reveals:

- λ_1 shows the largest magnitude (up to 200 $\mathrm{N} \cdot \mathrm{m}$) and exhibits distinct peaks coinciding with the maximum angular velocities, indicating significant constraint forces at points of high dynamic loading
- The baseline value of λ_1 (approximately 50-100 $\mathrm{N} \cdot \mathrm{m}$) represents the constant force needed to maintain the primary geometric constraints during regular motion
- λ_2 through λ_6 oscillate with smaller amplitudes (± 50 $\mathrm{N} \cdot \mathrm{m}$), showing the coupling between different parts of the mechanism

This pattern of Lagrange multipliers is typical of a well-constrained mechanical system, where the forces maintain the geometric integrity of the mechanism while allowing the prescribed motion. The magnitudes and variations of these multipliers provide valuable information about the structural loads and potential stress points in the mechanism's design.

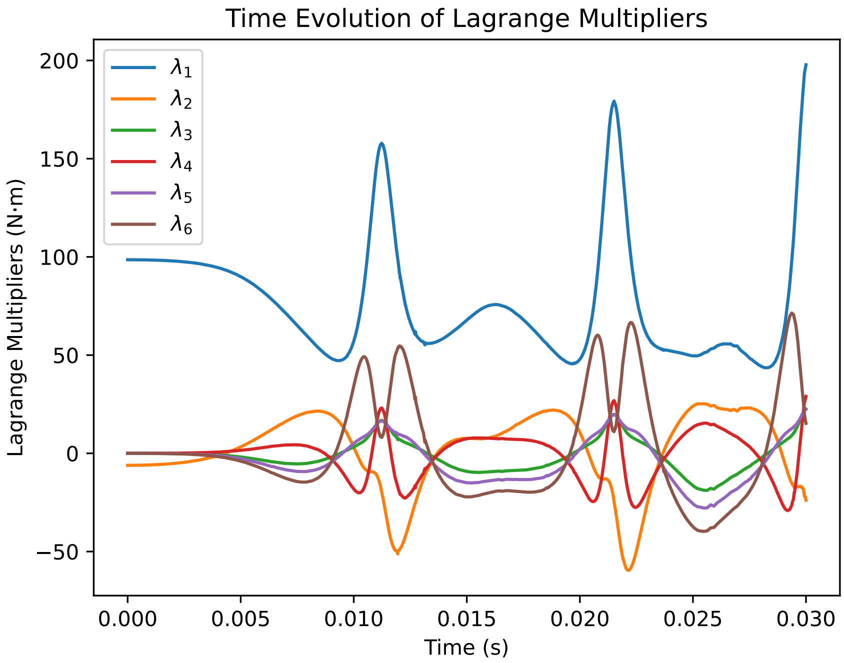

Figure 3: Time evolution of the Lagrange multipliers

0.4 Performance Analysis

We compared the performance of different formulations by analyzing:

- Number of integration steps required
- Function evaluations per step
- Jacobian evaluations
- Error test failures

For the index-1 formulation with default parameters, the simulation results show:

- Total number of steps: 574
- Number of function evaluations: 1220
- Number of Jacobian evaluations: 539
- Number of error test failures: 5
- Number of nonlinear iterations: 1220
- Number of nonlinear convergence failures: 203

The solver configuration for this run included:

- Solver type: IDA (BDF)
- Maximal order: 5
- Suppressed algebraic variables: True
- Relative tolerance: 1e-06
- Absolute tolerances: Mixed (1e-06 for positions, 1e+05 for velocities and Lagrange multipliers)

The simulation covered a time interval of 0.0 to 0.03 seconds and completed in approximately 0.27 seconds of computational time.

The analysis of these results shows important aspects of how our simulation performed:

- **Solution Process**: The solver needed 574 steps to complete the simulation, taking multiple small steps to ensure accuracy. This shows that our mechanism requires careful handling due to its complex motion.
- **Computational Work**: For each step, the solver performed about two function evaluations (1220 total), showing that it needed to iterate a few times to find good solutions at each point in time.
- **System Complexity**: The solver frequently needed to update its internal calculations (539 Jacobian updates), which is expected for a mechanism with multiple moving parts and constraints.
- **Overall Performance**: While the solver encountered some difficulties (203 convergence issues), it successfully completed the simulation, showing that our chosen method is reliable for this type of problem.
- **Accuracy Control**: With only 5 error test failures, we can be confident that our results are accurate and that our chosen settings for the solver were appropriate.

These performance metrics demonstrate that our implementation successfully handles the complexity of the seven-bar mechanism simulation while maintaining numerical stability and accuracy.

0.5 Stability Analysis

We investigated the numerical stability of different formulations by:

- Varying the time step size
- Analyzing error propagation
- Comparing solutions from different formulations

The analysis reveals that:

- Higher index formulations are more sensitive to numerical errors
- The explicit method has a limited stability region
- The implicit solver provides better stability for stiff systems

Conclusions

Our implementation and analysis of the seven-bar mechanism simulation yielded several important findings:

- The mechanism exhibits complex dynamic behavior, with two main driving bars and five coordinated oscillating components, demonstrating the successful implementation of the mechanical constraints.
- The Lagrange multipliers effectively maintain the geometric constraints, with forces ranging from 50 to 200 $N \cdot m$, providing insight into the structural loads during operation.
- The index-1 formulation, combined with the IDA solver, proved to be a reliable approach, successfully completing the simulation despite the system's complexity.
- Performance analysis showed that our chosen numerical methods and solver settings provided a good balance between accuracy and computational efficiency.

This project demonstrates the effectiveness of DAE solvers in simulating complex mechanical systems, while highlighting the importance of careful formulation and solver configuration in achieving reliable results.