**DYNAMICS** – Course Project

# Modelling, Control and Simulation of an Inverted Pendulum on a Cart

**Mary Sam** | October 2026

## 1 Introduction and objectives

The inverted pendulum on a cart is the classic benchmark for unstable, under-actuated systems; the same balancing problem appears in two-wheeled robots, self-balancing vehicles and rockets during launch. In this project a PID controller acting on the cart force keeps a 6.64 kg steel pendulum upright on a 75.3 kg cart. The objectives were to:

1. derive the nonlinear equations of motion with the Lagrangian method and linearise them about the upright equilibrium;
2. design a PID controller for the linearised model;
3. simulate the closed loop in MATLAB and Simulink;
4. build the system in SolidWorks, simulate it in Simscape Multibody with the same gains, and compare the CAD-based response with the analytical one.

## 2 System and CAD model

The system (Fig. 1) consists of a fixed platform with two rails, a steel cart on four wheels, and a pendulum made of a rod, a 100 mm steel sphere and a pivot pin, hinged on top of the cart. All parameters were taken from the SolidWorks mass properties (Table 1). The platform is fixed to the ground, so its mass does not enter the dynamics and is excluded. In the assembly the pendulum is not exactly vertical: it leans $\theta_0 = 0.02385$ rad (1.37°) towards $+x$. This tilt of the CAD pose is the **initial condition** of every simulation: the pendulum is released from 1.37° with the cart at rest, and the controller must bring it back to upright ($\theta = 0$).

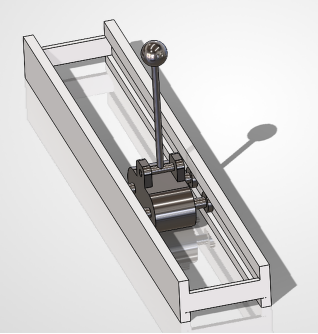


**Figure 1.** SolidWorks assembly of the inverted pendulum on a cart: rail platform, cart on four wheels and pendulum (rod and sphere) hinged on the cart. $x$ is the cart position along the rail, $\theta$ the pendulum tilt from vertical and $F$ the control force on the cart. In this pose the pendulum leans $\theta_0 = 1.37°$, which is used as the initial tilt in all simulations.

**Table 1.** Model parameters obtained from the CAD model (plain carbon steel, $\rho = 7800\,\mathrm{kg/m^3}$).

| Symbol | Quantity | Value | Source |
|---|---|---|---|
| $m$ | Pendulum mass (rod + sphere + pin) | 6.640 kg | CAD |
| $l$ | Pivot to pendulum centre of mass | 0.36605 m | CAD |
| $I_{cm}$ | Pendulum inertia about its centre of mass | 0.25241 kg m$^2$ | CAD |
| $J$ | Pendulum inertia about the pivot ($I_{cm} + ml^2$) | 1.14218 kg m$^2$ | Parallel axis |
| $m_c$ | Cart body with axles | 71.453 kg | CAD |
| $m_w$ / $I_w$ / $R_w$ | Wheel mass / inertia / radius | 0.588 kg / $5.88\times10^{-4}$ kg m$^2$ / 0.04 m | CAD |
| $M$ | Effective cart mass (rolling wheels, Sec. 3) | 75.275 kg | Calculated |
| $g$ | Gravity | 9.81 m/s$^2$ | – |
| $b$, $c$ | Rail and pivot friction | 0 | Assumption |
| $\theta_0$ | Initial tilt (CAD pose) | 0.02385 rad (1.37°) | CAD |

## 3 Analytical modelling (Phase 1)

**Assumptions.** The bodies are rigid with the CAD mass properties; motion is planar; there is no friction in the rail or pivot; the wheels roll without slip; the actuator applies the force directly to the cart and the angle is measured exactly (no delay or noise).

**Coordinates and energies.** The generalised coordinates are the cart position $x$ and the pendulum angle $\theta$ from the upright. The pendulum centre of mass is at $(x + l\sin\theta,\ l\cos\theta)$, so

$$T = \tfrac{1}{2}(M+m)\dot{x}^2 + ml\,\dot{x}\,\dot{\theta}\cos\theta + \tfrac{1}{2}J\dot{\theta}^2, \qquad V = mgl\cos\theta, \qquad \mathcal{L} = T - V. \tag{1}$$

**Lagrange's equations.** Applying $\dfrac{d}{dt}\Big(\dfrac{\partial \mathcal{L}}{\partial \dot{q}_i}\Big) - \dfrac{\partial \mathcal{L}}{\partial q_i} = Q_i$ with $Q_x = F$ and $Q_\theta = 0$ gives the coupled nonlinear equations of motion

$$(M+m)\ddot{x} + ml\cos\theta\,\ddot{\theta} - ml\,\dot{\theta}^2\sin\theta = F, \tag{2}$$

$$ml\cos\theta\,\ddot{x} + J\ddot{\theta} - mgl\sin\theta = 0. \tag{3}$$

**Linearisation.** About the upright equilibrium ($\sin\theta \approx \theta$, $\cos\theta \approx 1$, $\dot{\theta}^2 \approx 0$), equations (2)–(3) become $(M+m)\ddot{x} + ml\ddot{\theta} = F$ and $ml\ddot{x} + J\ddot{\theta} - mgl\,\theta = 0$. With state $\mathbf{s} = [x,\ \dot{x},\ \theta,\ \dot{\theta}]^{\mathsf{T}}$, input $F$, output $\mathbf{y} = [x,\ \theta]^{\mathsf{T}}$ and $\Delta = (M+m)J - (ml)^2 = 87.654\,\mathrm{kg^2 m^2}$:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{(ml)^2 g}{\Delta} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{(M+m)mgl}{\Delta} & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -0.6613 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 22.284 & 0 \end{bmatrix}, \qquad B = \begin{bmatrix} 0 \\ \frac{J}{\Delta} \\ 0 \\ -\frac{ml}{\Delta} \end{bmatrix} = \begin{bmatrix} 0 \\ 0.013031 \\ 0 \\ -0.027731 \end{bmatrix}, \tag{4}$$

$$\dot{\mathbf{s}} = A\mathbf{s} + BF, \qquad \mathbf{y} = C\mathbf{s} + DF, \qquad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, \quad D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \tag{5}$$

The pendulum-angle transfer function is

$$G_\theta(s) = \frac{\Theta(s)}{F(s)} = \frac{-0.02773}{s^2 - 22.28}, \qquad \text{poles } s = \pm 4.72 \text{ rad/s (plus a double pole at 0 for the cart).} \tag{6}$$

The pole at $+4.72\,\mathrm{rad/s}$ makes the upright position unstable, so feedback control is required. The derivation was verified with the Symbolic Math Toolbox (symbolic and numerical $A$, $B$ agree to $4 \times 10^{-15}$).

**Check of the CAD quantities.**

- $l = 0.36605\,\mathrm{m}$ (pivot at $0.5000\,\mathrm{m}$, centre of mass at $0.1339\,\mathrm{m}$ along the rod axis). Hand check: a $100\,\mathrm{mm}$ steel sphere ($4.084\,\mathrm{kg}$) at $451.6\,\mathrm{mm}$ from the pivot plus the rest of the mass as a uniform rod gives $l = 0.3647\,\mathrm{m}$, within $0.4\,\%$.
- $J = I_{cm} + ml^2 = 0.25241 + 6.6404 \times 0.36605^2 = 1.14218\,\mathrm{kg\,m^2}$. Shifting $I_{cm}$ to the part origin with the same theorem gives $0.371554\,\mathrm{kg\,m^2}$, matching SolidWorks' output-frame value ($0.371553\,\mathrm{kg\,m^2}$).
- $M = 71.453 + 4(0.5881 + 5.881 \times 10^{-4}/0.04^2) = 75.275\,\mathrm{kg}$: rolling without slip adds $I_w/R_w^2 = 0.368\,\mathrm{kg}$ per wheel.

## 4 Control design (Phase 2)

A parallel PID controller with a filtered derivative acts on the angle error $e = 0 - \theta$:

$$F(s) = \Big(K_p + \frac{K_i}{s} + \frac{K_d\,s}{T_f s + 1}\Big)E(s), \qquad T_f = 0.01\ \mathrm{s}\ (N = 100\ \mathrm{rad/s}). \tag{7}$$

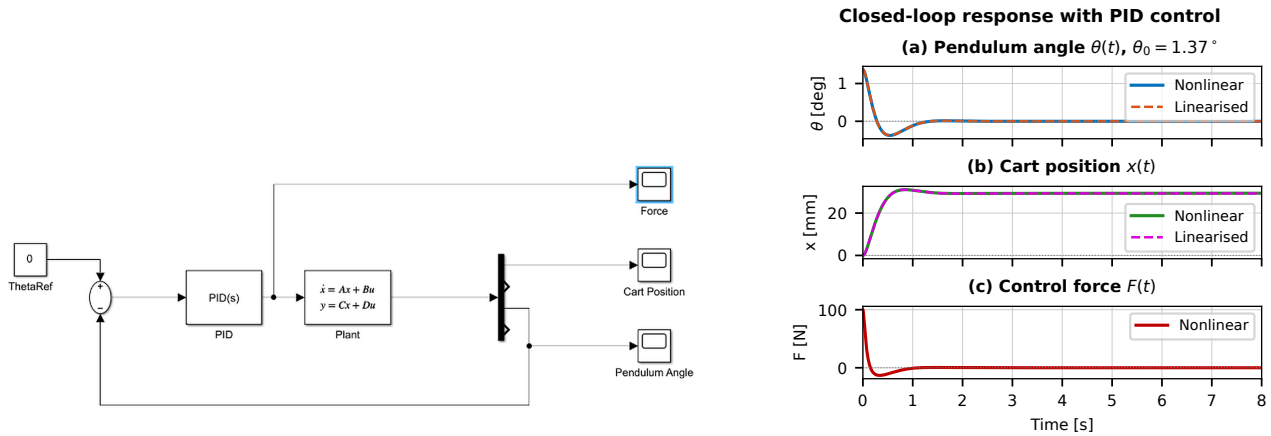
The gains are negative because $G_\theta(s)$ has a negative gain: pushing the cart forward tips the pendulum backward. The gains were tuned to minimise the ITAE of $\theta$ subject to a phase margin of at least $60°$, a peak sensitivity $M_s \le 1.2$ and a peak force of at most $100\,\mathrm{N}$ for the CAD tilt (Table 2). The main script also runs MATLAB's PID Tuner algorithm (`pidtune`) on the same plant at the same crossover frequency and phase margin as a cross-check.

**Table 2.** PID gains and loop properties.

| Parameter | Value | Property | Value |
|---|---|---|---|
| Proportional gain $K_p$ | $-4192.9$ | Phase margin | $60.0°$ at 17.95 rad/s |
| Integral gain $K_i$ | $-8365.6$ | Peak sensitivity $M_s$ | 1.16 |
| Derivative gain $K_d$ | $-645.6$ | Gain margin (gain reduction) | 0.30 ($-10.5$ dB) |
| Filter time constant $T_f$ | 0.01 s | Robustness | Stable for $\pm 20\,\%$ in $m$ and $M$ |

## 5 MATLAB and Simulink results

The closed loop was simulated from the CAD tilt $\theta_0 = 1.37°$ with the cart at rest, using the nonlinear equations (2)–(3) and the linearised model, and with the Simulink model `pendulum_pid.slx` (State-Space plant and PID Controller block, Fig. 2a).




**(a)** Simulink model with the PID control loop **(b)** Closed-loop response (analytical model)

**Figure 2.** Phase 2: (a) Simulink model `pendulum_pid.slx`; (b) pendulum angle, cart position and control force of the analytical model. The pendulum starts from the initial tilt $\theta_0 = 1.37^\circ$ of the CAD pose with the cart at rest; the nonlinear and linearised curves coincide.

**Table 3.** Closed-loop performance of the analytical model.

| Quantity | Value | Interpretation |
|---|---|---|
| Settling time of $\theta$ (2 %) | 1.22 s | Pendulum upright after 1.22 s |
| Undershoot past upright | 27.4 % ($-0.375^\circ$ at 0.54 s) | One small swing, no oscillation |
| Peak control force | 100.1 N | Largest push, at $t = 0$ |
| Cart peak / final position | 31.1 mm / 29.4 mm | Cart moves under the pendulum, then stops |
| Closed-loop poles | $-77.1$, $-16.7$, $-3.08 \pm 2.92j$ | All in the left half-plane: stable |

The pendulum balances: $\theta$ crosses zero at 0.28 s, swings $0.375^\circ$ past upright and stays within 2 % of $\theta_0$ after 1.22 s. The cart comes to rest 29.4 mm from its start once the pendulum is upright. The nonlinear and linearised angles differ by less than $2 \times 10^{-6}$ rad, and the Simulink model gives the same result.

## 6 Simscape model (Phase 3)

The SolidWorks assembly was exported with the Simscape Multibody Link plug-in and imported with `smimport` (Fig. 3). The platform is fixed to the world frame; the cart moves on a **prismatic joint** along the rail; the pendulum turns on a **revolute joint** at the pivot; the wheels turn freely on their axles. All joint damping is zero. The imported hinge reads $90^\circ$ when the pendulum is upright, so an offset of $-\pi/2$ converts it to $\theta$. The same PID Controller block and gains as in Phase 2 drive the prismatic joint, and the simulation starts from the CAD pose ($\theta_0 = 1.37^\circ$).

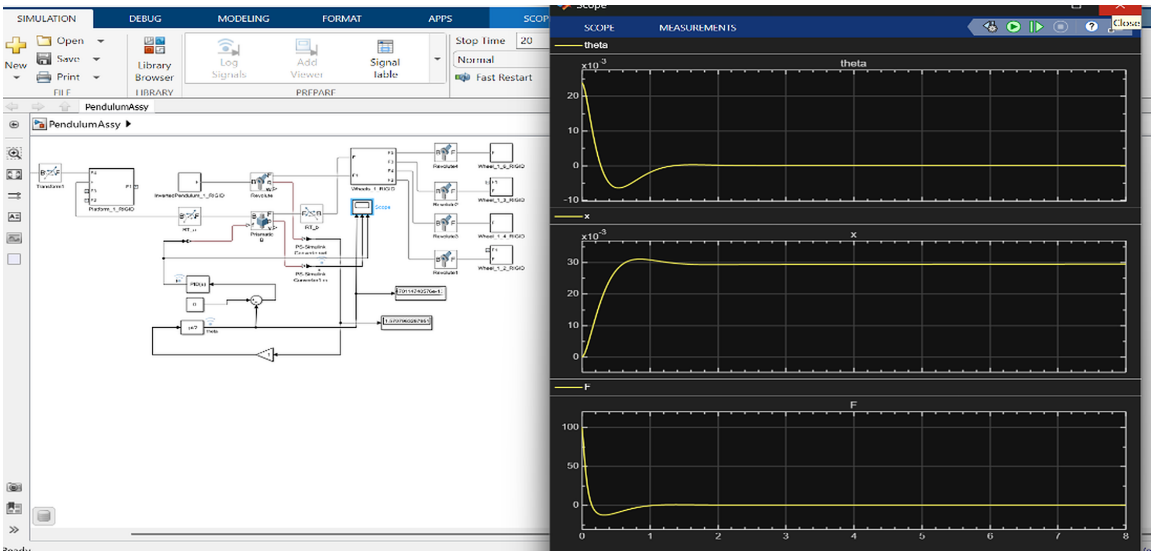



**Figure 3.** Phase 3: (a, left) CAD-based Simscape Multibody model `PendulumAssy.slx` with the PID control loop; (b, right) its scope output: pendulum angle $\theta$ [rad], cart position $x$ [m] and control force $F$ [N] over 8 s, starting from the same initial tilt $\theta_0 = 0.0239$ rad ($1.37^\circ$).

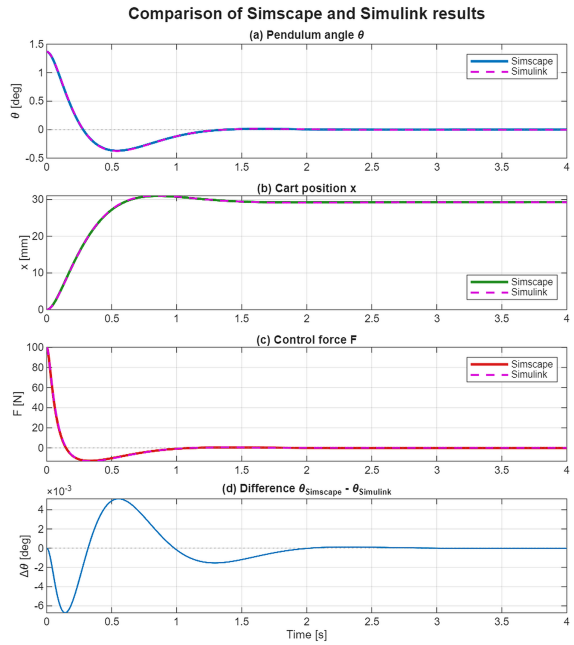



**Figure 4.** Comparison of the Simulink (analytical) and Simscape (CAD) results with identical PID gains: (a) pendulum angle, starting from the initial tilt $\theta_0 = 1.37°$, (b) cart position, (c) control force, (d) angle difference $\theta_{\text{Simscape}} - \theta_{\text{Simulink}}$.

**Table 4.** Analytical (Simulink) and CAD (Simscape) results.

| Metric | Simulink | Simscape | Difference | Difference (%) |
|---|---|---|---|---|
| Settling time of $\theta$, 2 % [s] | 1.219 | 1.224 | +0.005 | +0.41 |
| Undershoot [% of $\theta_0$] | 27.43 | 27.05 | −0.38 | −1.37 |
| Peak force [N] | 100.12 | 99.97 | −0.15 | −0.15 |
| Cart peak / final position [mm] | 31.05 / 29.26 | 30.98 / 29.41 | −0.07 / +0.15 | −0.21 / +0.51 |

## 7 Discussion

**Convergence and stability.** The open-loop model has an unstable pole at +4.72 rad/s. With the PID controller all closed-loop poles lie in the left half-plane, the phase margin is 60° and $M_s = 1.16$, so the loop is stable and robust: it stays stable for ±20 % changes in either mass. In both models $\theta$ converges to zero in about 1.22 s after a single undershoot, and the cart comes to rest.

**Differences between the analytical and CAD results.** The Simscape curve lies on top of the Simulink curve (Fig. 4). The largest angle difference is 0.0067° (RMS 0.0014°), the settling time differs by 5 ms and every metric in Table 4 agrees to within 1.4 %.

**Sources of discrepancy.** (i) *Wheel rolling inertia*: the analytical cart mass includes it ($M = 75.28$ kg), while the Simscape wheels spin freely without rail contact (73.80 kg); this is the most likely cause of the 0.38-point difference in undershoot. (ii) *Inertia details*: Simscape uses the full 3-D inertia tensors from CAD, the analytical model only the planar values. (iii) *Gravity constant*: 9.81 against 9.80665 m/s$^2$. (iv) *Linearisation*: the Simulink plant is linear; at 1.37° this contributes only about $10^{-6}$ rad. (v) *Solver tolerance*: ode45 against a variable-step Simscape solver (relative tolerance $10^{-6}$). (vi) *Zero friction*: both models neglect friction, so their agreement validates the equations and CAD parameters but not a physical rig.

**Limitations.** The controller uses only $\theta$, so the cart position is not regulated: after a disturbance the pendulum recovers within about 1 s but the cart keeps drifting. The heavy steel cart also needs about 100 N for a 1.4° tilt. Cart-position feedback (cascade PID or LQR) and a lighter cart would address both.

## 8 Conclusions

The Lagrangian model, built from CAD-derived parameters, has an unstable pole at +4.72 rad/s. A PID controller ($K_p = -4192.9$, $K_i = -8365.6$, $K_d = -645.6$) with a 60° phase margin balances the pendulum from a 1.37° tilt in 1.22 s with a peak force of 100 N, and the cart settles 29 mm from its start. The CAD-based Simscape model, with the same gains, agrees with the analytical model to within 0.0067° in angle and 1.4 % in all metrics.

**References.** [1] MathWorks, *Control System Toolbox: pidtune / PID Tuner*, MATLAB R2025b. [2] MathWorks, *Simscape Multibody Link and smimport*, MATLAB R2025b. [3] K. Ogata, *Modern Control Engineering*, 5th ed., Prentice Hall, 2010.

## Appendix: project files and how to reproduce the results

All files are in the submitted folder. They need **MATLAB R2025b** with Simulink, Control System Toolbox, Symbolic Math Toolbox, Simscape and Simscape Multibody. SolidWorks 2026 is needed only to open or re-export the CAD model.

**Table 5.** Project files.

| File | Phase | Purpose |
|---|---|---|
| *MATLAB scripts* | | |
| `params.m` | All | All model parameters from the CAD model, and the linearised $A$, $B$, $C$, $D$ matrices. Called by the other scripts. |
| `derive_symbolic.m` | 1 | Lagrangian derivation, nonlinear equations of motion and linearisation with the Symbolic Math Toolbox; checks the result against `params.m`. |
| `main_pid_inverted_pendulum.m` | 2 | Transfer function, PID gains, loop margins, nonlinear and linear simulations and the 3-panel figure. Saves `pid_gains.mat` and `analytical_results.mat`. |
| `optimize_pid.m` | 2 | Optional: reproduces the PID gains (ITAE minimisation with margin and force constraints). Takes a few minutes. |
| `build_simulink_pid_model.m` | 2 | Builds and runs the Simulink model `pendulum_pid.slx`. |
| `compare_pid_vs_assy.m` | 3 | Runs `pendulum_pid.slx` and `PendulumAssy.slx`, plots both on the same axes and prints the comparison table. |
| *Models and data* | | |
| `pendulum_pid.slx` | 2 | Simulink model: linearised State-Space plant with the PID Controller block. |
| `PendulumAssy.slx` | 3 | Simscape Multibody model imported from SolidWorks, with prismatic and revolute joints and the PID loop. |
| `PendulumAssyDisturb.slx` | 3 | Same model with a torque pulse on the pendulum (disturbance test). |
| `PendulumAssy_DataFile.m` | 3 | Data written by `smimport` (joint frames, masses, inertias); needed by both Simscape models. |
| `PendulumAssy.xml`, `*_sldprt.STEP` | 3 | SolidWorks export (assembly structure and 3-D geometry of each part). |
| `pid_gains.mat`, `analytical_results.mat` | 2 | Saved gains and analytical simulation results (created by the main script). |
| *CAD files* | | |
| `Pendulum-Assy.SLDASM` | 3 | SolidWorks assembly (open together with the part files). |
| `Platform`, `Wheels`, `Wheel 1`, `InvertedPendulum .SLDPRT` | 3 | SolidWorks parts: rail platform, cart body, wheel and pendulum. |

### How to run

1. Copy the whole folder to your computer, open MATLAB and set the folder as the *Current Folder*.
2. **Phase 1:** run `derive_symbolic`. The command window shows the nonlinear equations, the $A$ and $B$ matrices and two norms close to $10^{-15}$.
3. **Phase 2:** run `main_pid_inverted_pendulum`. It prints the open-loop poles, the transfer function, the gains, the phase margin and the performance figures, and opens the 3-panel response plot (Fig. 2b).
4. Run `build_simulink_pid_model` (or open `pendulum_pid.slx` and press *Run*) to simulate the Simulink model; open the scopes to see $\theta$, $x$ and $F$.
5. **Phase 3:** run `PendulumAssy_DataFile`, then open `PendulumAssy.slx` and press *Run*. Mechanics Explorer shows the 3-D animation of the pendulum balancing; the scope shows $\theta$, $x$ and $F$.
6. **Comparison:** run `compare_pid_vs_assy`. It produces Fig. 4 (saved as `fig_simulink_vs_simscape.png`) and the values of Table 4 (saved as `simulink_vs_simscape.csv`).
7. **Optional:** run `optimize_pid` to re-derive the gains, or open `PendulumAssyDisturb.slx` and press *Run* for the disturbance test.