

Guía ABP: Simulación Web — Competencia de Rodadura en un Plano Inclinado

Facultad de Ingeniería – Universidad de Antioquia

Abstract

Documento orientador para un proyecto de Aprendizaje Basado en Problemas (ABP). Los estudiantes, con apoyo de IA, bibliografía y recursos tecnológicos, deben diseñar una **simulación HTML** que represente la caída simultánea por un mismo plano inclinado de tres cuerpos: *esfera maciza*, *cilindro macizo* y *aro (cilindro hueco)*. El objetivo es entender y demostrar, tanto numérica como visualmente, cuál objeto llega primero a la base y por qué, apoyándose en la física rotacional y la condición de rodadura sin deslizamiento.

Objetivos de aprendizaje

- Comprender y aplicar las ecuaciones de la dinámica de cuerpos rígidos en rodadura sin deslizamiento.
- Diseñar una simulación interactiva en HTML/JavaScript que muestre el comportamiento dinámico de cuerpos rodantes.
- Usar herramientas digitales (p.ej. p5.js, matter.js, bibliotecas JS) y recursos en línea para implementar y validar modelos físicos.
- Comunicar resultados y comparar teoría con simulación y experimentación virtual.

Recursos sugeridos (tecnológicos y formativos)

- **Librerías JavaScript:** *p5.js* para visualización y control del loop gráfico; *matter.js* o *cannon.js* como motores de física si se desea comportamiento de colisiones y cuerpos rígidos avanzado. (Ver tutoriales introductorios y de integración p5.js + matter.js). :contentReference[oaicite:1]index=1
- **Tutoriales en vídeo:** hay múltiples guías que muestran cómo codificar simulaciones físicas en JS y específicamente comparativas de objetos rodando por un plano inclinado (sphere vs ring vs disc). Estos vídeos son útiles para obtener ideas sobre la implementación y la visualización. :contentReference[oaicite:2]index=2

- **Lectura técnica:** capítulos de dinámica rotacional en libros de física universitaria (ej. Marion & Thornton, Kleppner & Kolenkow, Serway & Jewett) para fórmulas de momento de inercia y condiciones de rodadura.
- **IA y asistentes:** usar ChatGPT u otras IA para generar pseudocódigo, depurar errores y explicar pasos matemáticos.
- **Herramientas de desarrollo:** Overleaf (para el informe), editor web (VS Code, p5 web editor), navegador moderno (Chrome/Firefox) para probar HTML/JS.

Enunciado del problema (para los estudiantes)

Diseñar una simulación web que muestre tres objetos que ruedan sin deslizar desde la misma altura inicial sobre el mismo plano inclinado y que compitan para llegar a la base. Los objetos son:

- (a) Esfera maciza (momento de inercia $I = \frac{2}{5}mR^2$).
- (b) Cilindro macizo (disco) (momento de inercia $I = \frac{1}{2}mR^2$).
- (c) Aro o cilindro hueco (momento de inercia $I = mR^2$).

Requisitos de la simulación:

- Los tres cuerpos deben partir desde reposo a la misma altura y al mismo tiempo.
- La simulación debe mostrar posición, velocidad lineal y velocidad angular de cada cuerpo en tiempo real (texto o gráficos).
- La interfaz debe permitir ajustar el ángulo del plano θ , la altura inicial h y, opcionalmente, el radio R y la masa m (nota: para la teoría de rodadura sin deslizamiento la masa no afecta la aceleración si la forma geométrica se mantiene).
- Se debe incluir un botón "Iniciar competencia"; un "Reset", opcionalmente, la opción de mostrar vectores de fuerzas y torques.
- Acompañar la simulación con un informe que incluya: deducción matemática, predicción teórica de tiempos y aceleraciones, comparación con los resultados de la simulación y discusión de discrepancias.

Guía pedagógica (pasos para los estudiantes)

1. **Revisión teórica:** repasar rodadura sin deslizamiento, relación entre aceleración lineal a y aceleración angular α : $a = \alpha R$, y la segunda ley rotacional $\tau = I\alpha$.
2. **Deducción matemática** (ver sección siguiente).
3. **Prototipo visual:** crear un boceto de la interfaz (controles, vista lateral del plano inclinado, indicadores).

4. **Elección tecnológica:** decidir si usar p5.js para dibujar y actualizar estado y matter.js (o cannon.js) para dinámica, o implementar la física manualmente con integrador simple (Euler semi-implícito recomendado).
5. **Implementación incremental:**
 - (a) Implementar el dibujo del plano y un cuerpo simple que se mueva con aceleración constante (sin rotación).
 - (b) Añadir rotación y condición de rodadura: acoplar $v = \omega R$.
 - (c) Añadir los tres cuerpos y las comparaciones gráficas numéricas.
 - (d) Validar con la predicción teórica.
6. **Validación y documentación:** ejecutar varios casos (distintos θ , R), registrar tiempos, elaborar gráficos y conclusiones.

Desarrollo matemático (científico y claro)

Planteamiento

Considere un objeto rígido de masa m y radio R que rueda sin deslizar sobre un plano inclinado con ángulo θ . La componente de la gravedad paralela al plano es $mg \sin \theta$. Sea f la fuerza de fricción estática que produce el torque necesario para la rotación (no realiza trabajo negativo neto en la rodadura ideal).

Ecuaciones

Traslación del centro de masa:

$$ma = mg \sin \theta - f.$$

Rotación (sobre el centro de masa, torque por fricción):

$$fR = I\alpha.$$

Condición de rodadura sin deslizamiento:

$$a = \alpha R.$$

Sustituyendo $\alpha = a/R$ en la ecuación rotacional:

$$fR = I \frac{a}{R} \quad \Rightarrow \quad f = \frac{I}{R^2} a.$$

Sustituir en la ecuación traslacional:

$$ma = mg \sin \theta - \frac{I}{R^2} a$$

$$\Rightarrow a \left(m + \frac{I}{R^2} \right) = mg \sin \theta$$

$$a = \frac{g \sin \theta}{1 + \frac{I}{mR^2}}$$

Aplicación a los tres cuerpos

Usamos la fórmula anterior con el I correspondiente.

- **Aro (cilindro hueco):** $I = mR^2$.

$$a_{\text{aro}} = \frac{g \sin \theta}{1 + 1} = \frac{g \sin \theta}{2}.$$

- **Cilindro macizo (disco):** $I = \frac{1}{2}mR^2$.

$$a_{\text{cil}} = \frac{g \sin \theta}{1 + \frac{1}{2}} = \frac{2}{3}g \sin \theta.$$

- **Esfera maciza:** $I = \frac{2}{5}mR^2$.

$$a_{\text{esf}} = \frac{g \sin \theta}{1 + \frac{2}{5}} = \frac{5}{7}g \sin \theta.$$

Numéricamente, comparando factores multiplicativos de $g \sin \theta$:

$$\frac{5}{7} \approx 0.714, \quad \frac{2}{3} \approx 0.667, \quad \frac{1}{2} = 0.5.$$

Por tanto, **la esfera maciza acelera más rápido**, luego el cilindro macizo y por último el aro. En igualdad de condiciones geométricas (igual radio y altura inicial), **la esfera llegará primero a la base**. Esta predicción se puede corroborar visualmente con la simulación. (Ver explicaciones pedagógicas en vídeos y demostraciones prácticas). :contentReference[oaicite:3]index=3

Sugerencias de implementación técnica (HTML/JS)

Opción A — p5.js + física manual (recomendado para control didáctico)

- Use p5.js para dibujar y controlar el loop (`setup()` y `draw()`).
- Mantenga un estado por objeto: posición a lo largo del plano $s(t)$, velocidad $v(t)$, velocidad angular $\omega(t) = v/R$.
- Actualice con un integrador (recomendado: **Euler semi-implícito** o **RK2**) para estabilidad.

Opción B — matter.js (motor de física)

- matter.js facilita colisiones y cuerpos rígidos 2D; permite simular objetos con fricción y restricciones.
- Si se usa un motor, asegúrese de implementar la condición de rodadura o de ajustar parámetros de fricción y momento de inercia de cada cuerpo (si el motor lo permite).
- Revisar tutoriales para integrar matter.js con p5.js o canvas directamente. :contentReference[oaicite:4]index=4

Controles y visualización recomendada

- Panel lateral con sliders: ángulo θ , altura h , radio R (opcional).
- Etiquetas con valores en tiempo real: v , ω , a , tiempo transcurrido t .
- Dibujar flechas vectoriales que representen fuerzas (peso, reacción normal y fricción) para apoyo pedagógico.

Actividades obligatorias y producto final

1. **Código de la simulación** (repositorio o archivo HTML/JS/CSS).
2. **Informe en Overleaf** (máx. 6 páginas) con:
 - Deducción teórica y predicciones numéricas.
 - Capturas de la simulación, tablas de tiempos y gráficos comparativos.
 - Análisis de discrepancias y posibles fuentes de error (integrador, parámetros del motor, fricción parásita).
3. **Video corto (2–4 min)** mostrando la simulación y comentando resultados.

Criterios de evaluación (ejemplo)

- Correctitud física (40%): derivaciones y concordancia con la teoría.
- Calidad de la simulación (30%): interfaz, robustez, controles y visualizaciones.
- Análisis y comunicación (20%): informe claro, gráficos y conclusiones.
- Uso de recursos y documentación (10%): referencias, uso de IA y tutoriales, estilo del código.

Ayudas y pistas para los estudiantes

- Emplear IA para generar pseudocódigo y depurar errores no conceptuales (p.ej. sintaxis JS).
- Consultar videos sobre cómo usar p5.js y matter.js como punto de partida. :contentReference[oaicite:5]index=5
- Si se usa un motor (matter.js), ver cómo ajustar el momento de inercia y la fricción para cada cuerpo; si el motor no permite variaciones precisas del momento de inercia, implementar la física manualmente.
- Probar casos extremos: θ pequeño (cercano a 0) y θ grande (ej. 45°) para ver concordancia con las predicciones.

Extensiones opcionales (para estudiantes avanzados)

- Añadir rozamiento cinético y estudiar cuándo se pierde la condición de rodadura.
- Simular cuerpos con distribución de masa no uniforme (Ej.: cilindro con eje descentrado).
- Implementar una versión 3D con Three.js + Cannon.js o Ammo.js.

Referencias y tutoriales útiles

- Tutoriales p5.js (introducción a la programación gráfica en JS). :contentReference[oaicite:6]index=6
- Introducción a matter.js (motor de física 2D, instalación y ejemplos). :contentReference[oaicite:7]index=7
- Videos demostrativos comparando esfera, disco y aro en un plano inclinado (explicación física y experimentos). :contentReference[oaicite:8]index=8