Universidad Rey Juan Carlos

Máster Universitario en Informática Gráfica, Videojuegos y Realidad Virtual

Curso Académico 2019/2020

Solvers de Elasticidad y Restricciones en Simulación

Autora: Clara Peñalva Carbonell

Tutor: Miguel Ángel Otaduy Tristán

Índice

1. Motivación y objetivos

1. Motivación y objetivos

- La animación basada en física se utiliza en distintos sectores:
  - Industria del cine
  - Videojuegos
  - Simuladores de entrenamiento
  - …

  Distintas necesidades de mercado

- Requisitos de las técnicas de animación:
  - Rápida
  - Precisa
  - General
  - Simple

  No existe un método perfecto

1. Motivación y objetivos

- Analizar métodos de integración:
  - Position-Based
  - Projective Dynamics

2 perspectivas

Estática

Dinámica

2. Conceptos previos

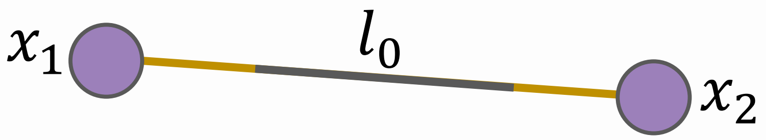
2.1 Backward Euler Implícito

- Segunda Ley de Newton: $\sum F_i = m \cdot a$
- Posiciones a lo largo del tiempo:
  - $v(t + dt) = v(t) + M^{-1} F\left(x(t + dt)\right) \cdot dt$
  - $x(t + dt) = x(t) + v(t + dt) \cdot dt$
- Linealizando fuerzas:
  - $\left(M - dt^2 \cdot \frac{dF}{dx}\right) \cdot v(t + dt) = M \cdot v(t) + dt \cdot F(x(t))$

2.2 Propiedad C

- C = propiedad del sistema que se quiere mantener
- Ejemplo: en masa-muelle → métrica de deformación:

$$C = |x_1 - x_2| - l_0$$



- Energía: $E = 1/2 \cdot k \cdot C^T \cdot C$ → fuerzas, Hessiana…

- Restricción → Débil / Fuerte → Optimización

3. Escenario de prueba

3. Escenario de prueba

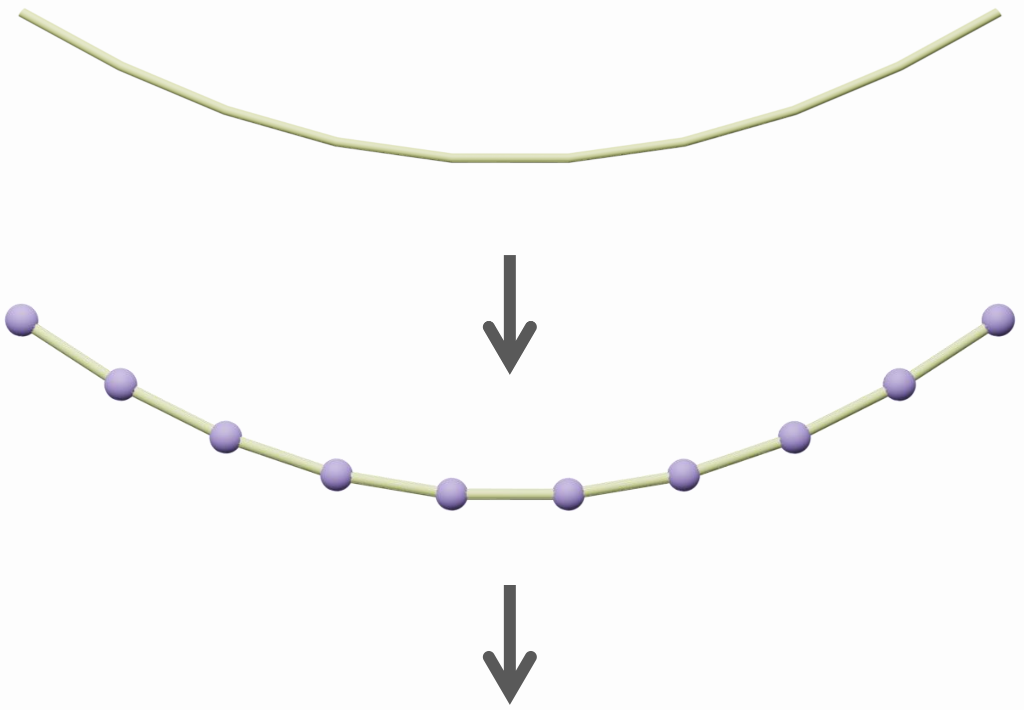
¿Qué queremos?

- Simular objeto deformable
- Repetibilidad de las pruebas
- Resultados manejables
- Escenario controlable

¿Cómo lo conseguimos?

- Discretización masa-muelle
- Fuerzas de elasticidad entre nodos
- Simulamos una cuerda con extremos fijos
- Cae por la gravedad

3. Escenario de prueba



10 nodos y 9 muelles

¿Cómo lo conseguimos?

- Discretización masa-muelle
- Fuerzas de elasticidad entre nodos
- Simulamos una cuerda con extremos fijos
- Cae por la gravedad

3.1 Condiciones de contorno

- Establecer condiciones de contorno en el sistema lineal antes de su resolución:

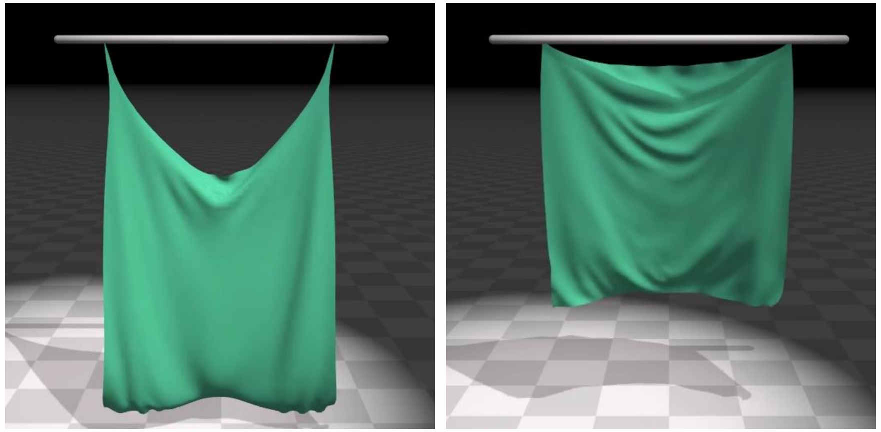
$$\begin{pmatrix} A_1 & A_2 \\ A_3 & A_4 \end{pmatrix} \begin{pmatrix} \mathrm{x}_1 \\ \mathrm{x}_2 \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix} \xrightarrow{\text{fijar } x_2 = \bar{x}_2} \begin{pmatrix} A_1 & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} b_1 - A_2 \bar{x}_2 \\ \bar{x}_2 \end{pmatrix}$$

- Vectores precalculados para redefinir A y b

4. Solvers

4.1 Position-Based Dynamics (Müller, et al., 2007)

- Resuelve directamente las posiciones
- C → **restricción** proyectada por cada muelle
- Resultados visualmente plausibles ✓
- Eficiente ✓
- La rigidez no tiene una base física X
- La rigidez depende del número de iteraciones X

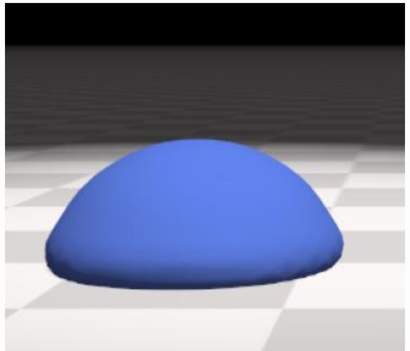


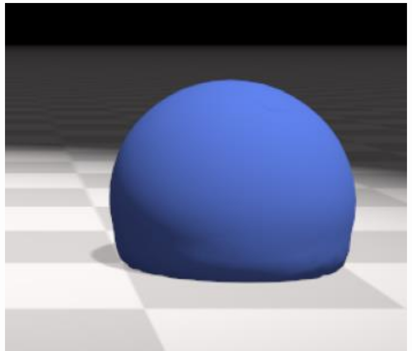
20 iteraciones 160 iteraciones

(Macklin, et al., 2016)

4.1.1 Extended Position-Based Dynamics (Macklin, et al., 2016)

- Deriva de una formulación de *compliant constraint:*
  - *Compliance:* $\alpha = 1/k$ y $\tilde{\alpha} = \alpha/dt^2$ para cada restricción
- Introduce multiplicadores de Lagrange: $\lambda = -\tilde{\alpha}^{-1}C(x)$
- $F = -\frac{\partial E}{\partial x} = -\nabla C^T(x)\,\tilde{\alpha}^{-1}C(x) = -\nabla C^T(x)\,\lambda$





(Macklin, et al., 2016)

4.1.1 Extended Position-Based Dynamics (Macklin, et al., 2016)

- Ecuación de movimiento *compliant constrained*:

$$M a - \nabla C^T(x)\lambda = 0$$
$$C(x) + \alpha\lambda = 0$$

$$\Big\downarrow \text{ Aproximaciones}$$

$$\left[\nabla C(x_i) M^{-1} \nabla C(x_i)^T + \tilde{\alpha}\right]\Delta\lambda = -C(x_i) - \tilde{\alpha}\lambda_i$$

4.1.1 Extended Position-Based Dynamics (Macklin, et al., 2016)

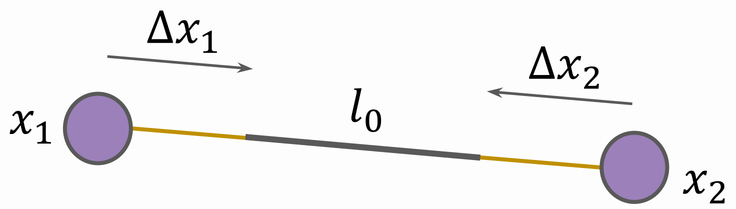
En general:

- Por cada restricción:
  - $\Delta\lambda = \frac{-C_j(x_i) - \widetilde{\alpha}_j \lambda_{ij}}{\nabla C_j M^{-1} \nabla C_j^T + \widetilde{\alpha}_j}$
  - $\Delta x = M^{-1} \nabla C(x_i)^T \Delta\lambda$

En nuestra prueba:

- Por cada restricción:
  - $\mathbf{C} = |x_1 - x_2| - l_0 = 0$
  - $\nabla C(x_1, x_2) = \begin{pmatrix} \frac{\partial C}{\partial x_1} & \frac{\partial C}{\partial x_2} \end{pmatrix}$

$$= \begin{pmatrix} \frac{x_1 - x_2}{|x_1 - x_2|} & \frac{-(x_1 - x_2)}{|x_1 - x_2|} \end{pmatrix}$$



4.2 Projective Dynamics

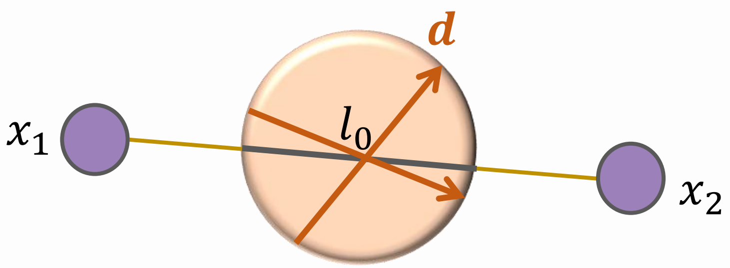
- Euler Implícito visto como una optimización:

$$x^{n+1} = \min_{x}\left(\frac{1}{2dt^2}\left(x^n - y\right)^T M\left(x^n - y\right) + E\left(x\right)\right)$$

4.2.1 Fast Simulation of Mass-Spring Systems (Liu, et al., 2013)

(Liu, et al., 2013)

- Ley de Hooke optimizada con restricciones:
  - $C = |d| = l_0$
  - $\frac{1}{2}k(|x_1 - x_2| - l_0)^2 = \frac{1}{2}k(|x_1 - x_2| - |d|)^2$



4.2.1 Fast Simulation of Mass-Spring Systems (Liu, et al., 2013)

$$\min_{x \in R^{3m}, d \in U} \frac{1}{2} x^T (M + dt^2 L) x - dt^2 x^T J d + x^T b$$

- Local: a cada energía, proyectando d en cada muelle
- Global: a todos los nodos, se resuelve un sistema lineal
- Constante: no depende de x o d, podemos precalcularlo

4.2.1 Fast Simulation of Mass-Spring Systems (Liu, et al., 2013)

$$\min_{x \in R^{3m}, d \in U} \ \frac{1}{2} x^T (M + dt^2 L) x - dt^2 x^T J d + x^T b$$

- **Local**: $\mathrm{d_i} = \mathrm{l_0} \frac{\mathrm{x_1} - \mathrm{x_2}}{|\mathrm{x_1} - \mathrm{x_2}|}$
- **Global**: $(M + dt^2 L) x = J \cdot d \cdot dt^2 - b$

El *solver* alterna los cálculos

- **Constante**: podemos precalcularlo

4.2.2 ADMM ⊇ Projective Dynamics (Overby, et al., 2017)

- Define variables locales a cada energía:
  - $z = Dx$

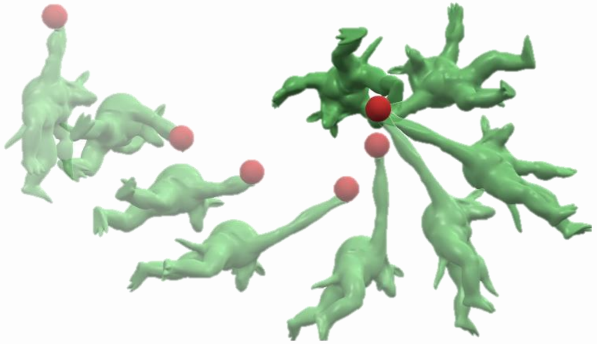
    z = variable auxiliar que define una restricción

    D = matriz de reducción → coordenadas locales

- Resolución lineal global

↓

Permite aplicar el *solver* ADMM



(Overby, et al., 2017)

4.2.2 ADMM ⊇ Projective Dynamics (Overby, et al., 2017)

- Una vez resuelto el problema por ADMM, queda:

- $\boldsymbol{z_i}^{\boldsymbol{n+1}} = \boxed{\underset{z_i}{arg\,min}}\left(U_i(z_i) + \frac{1}{2}|W_i(D_i x^{n+1} - z_i + \bar{u}_i{}^n)|^2\right)$
- $\overline{\boldsymbol{u_i}}^{\boldsymbol{n+1}} = \bar{u}_i{}^n + D_i x^{n+1} - z_i{}^{n+1}$

Variable dual de ADMM

Local

- $(M + dt^2 D^T W^T W D) \cdot \boldsymbol{x}^{\boldsymbol{n+1}} = \left(M\tilde{x} + dt^2 D^T W W(z^n - \bar{u}^n)\right)$

Global

4.2.2 ADMM ⊇ Projective Dynamics (Overby, et al., 2017)

- En nuestra prueba:

$$z_i^{n+1} = \frac{1}{2} \cdot \left(p_i^{n+1} + D_i \cdot x^{n+1} + \overline{u_i}^n\right)$$

$$P_i \leftarrow proj_{C_i}\,(D_i x)$$

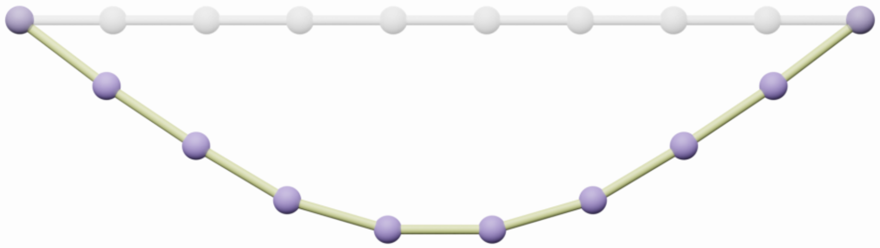
5. Estudio y discusión de resultados

5.1 Introducción

- Métricas:
  - Análisis estático:
    - Error absoluto del estado de reposo con respecto a la solución base (backward Euler Implícito con 5 iteraciones, dt: 0.001s)
  - Análisis dinámico:
    - Oscilaciones (máximos de E_k)
    - Periodo de E_k
- Consideramos que llega a reposo cuando: $E_k < 0{,}001J$

5.1 Introducción

- 2 casos de rigidez:

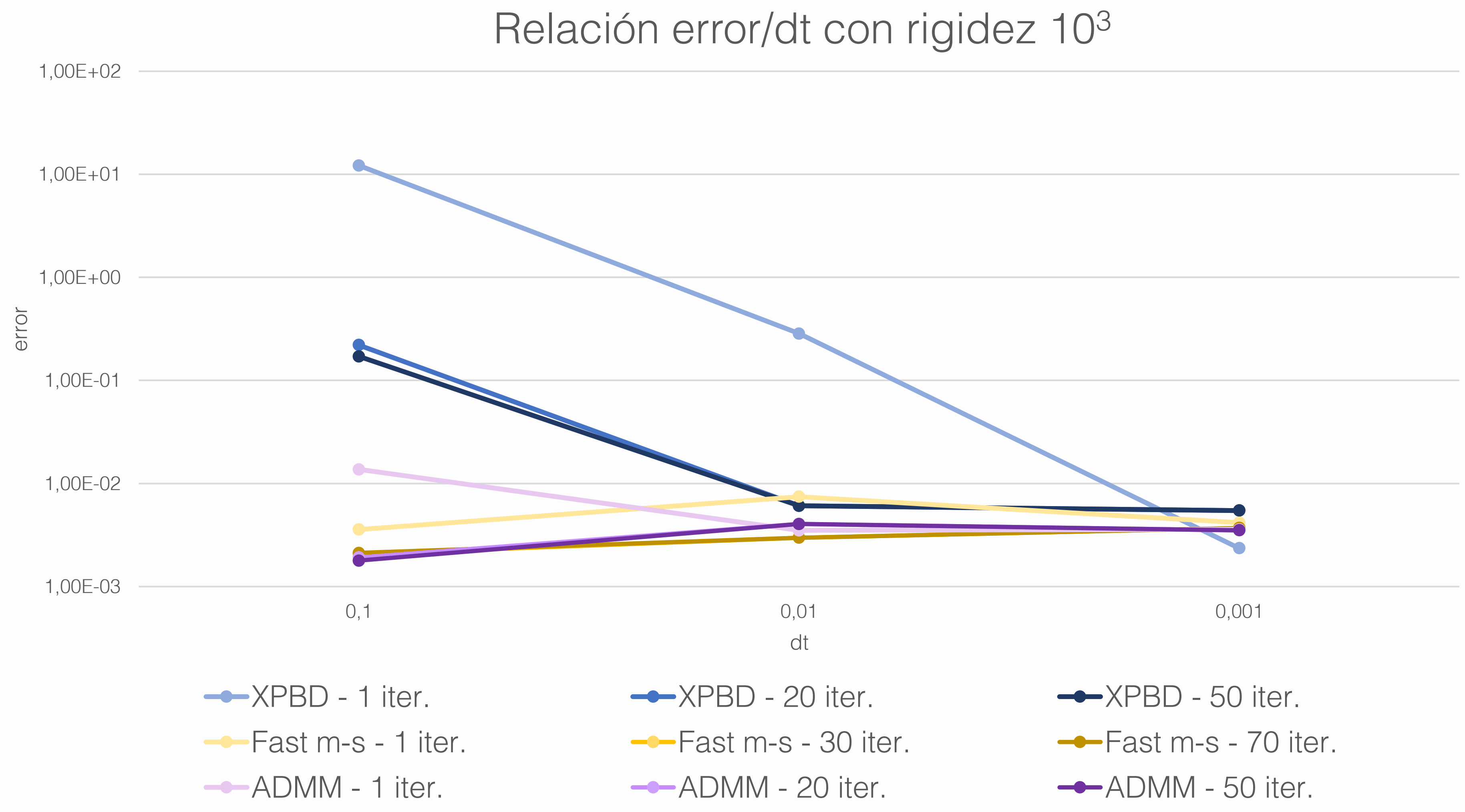


Rigidez 10^3

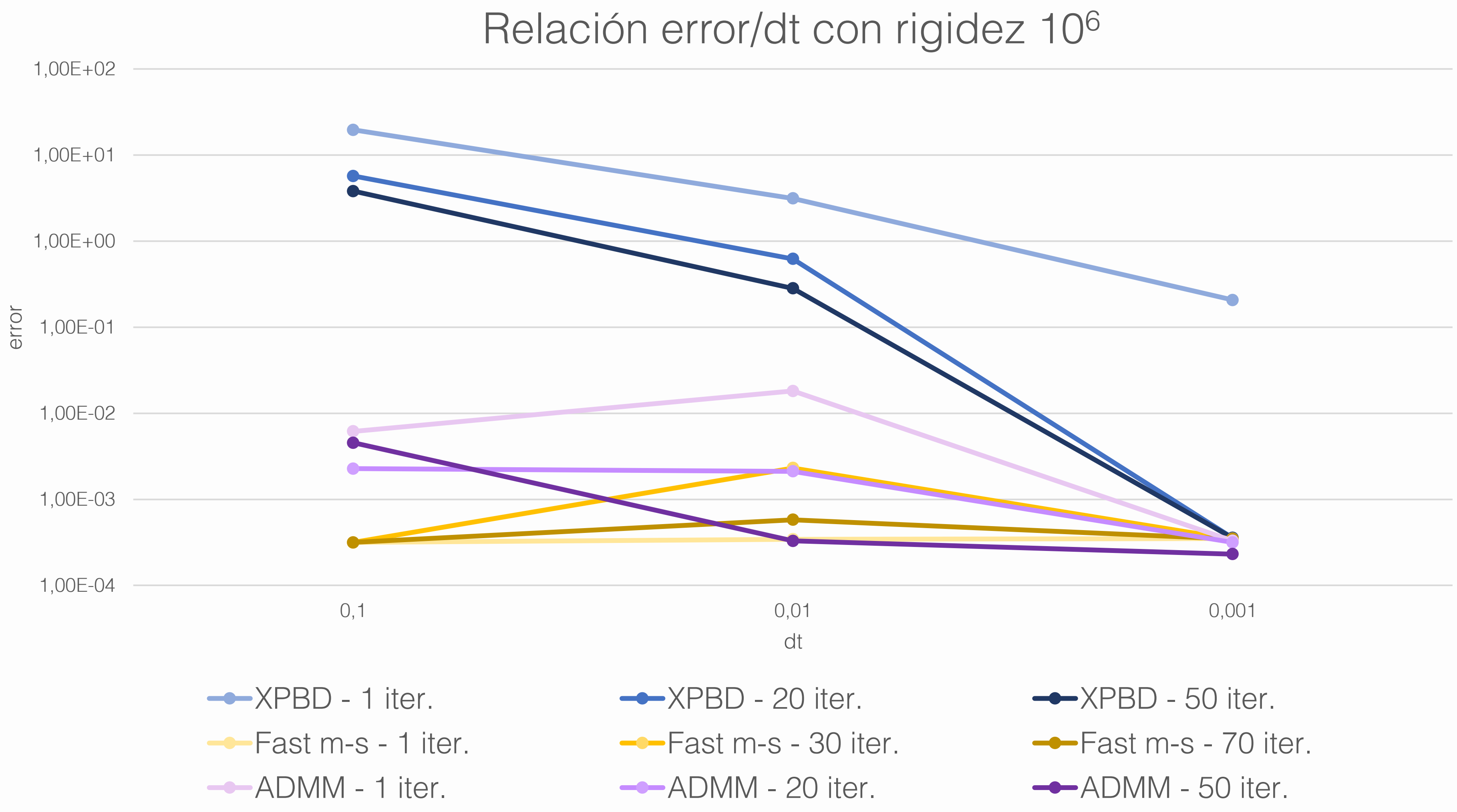
Rigidez 10^6

- 3 muestras de *dt* e iteraciones

5.2 Estudiando la estática



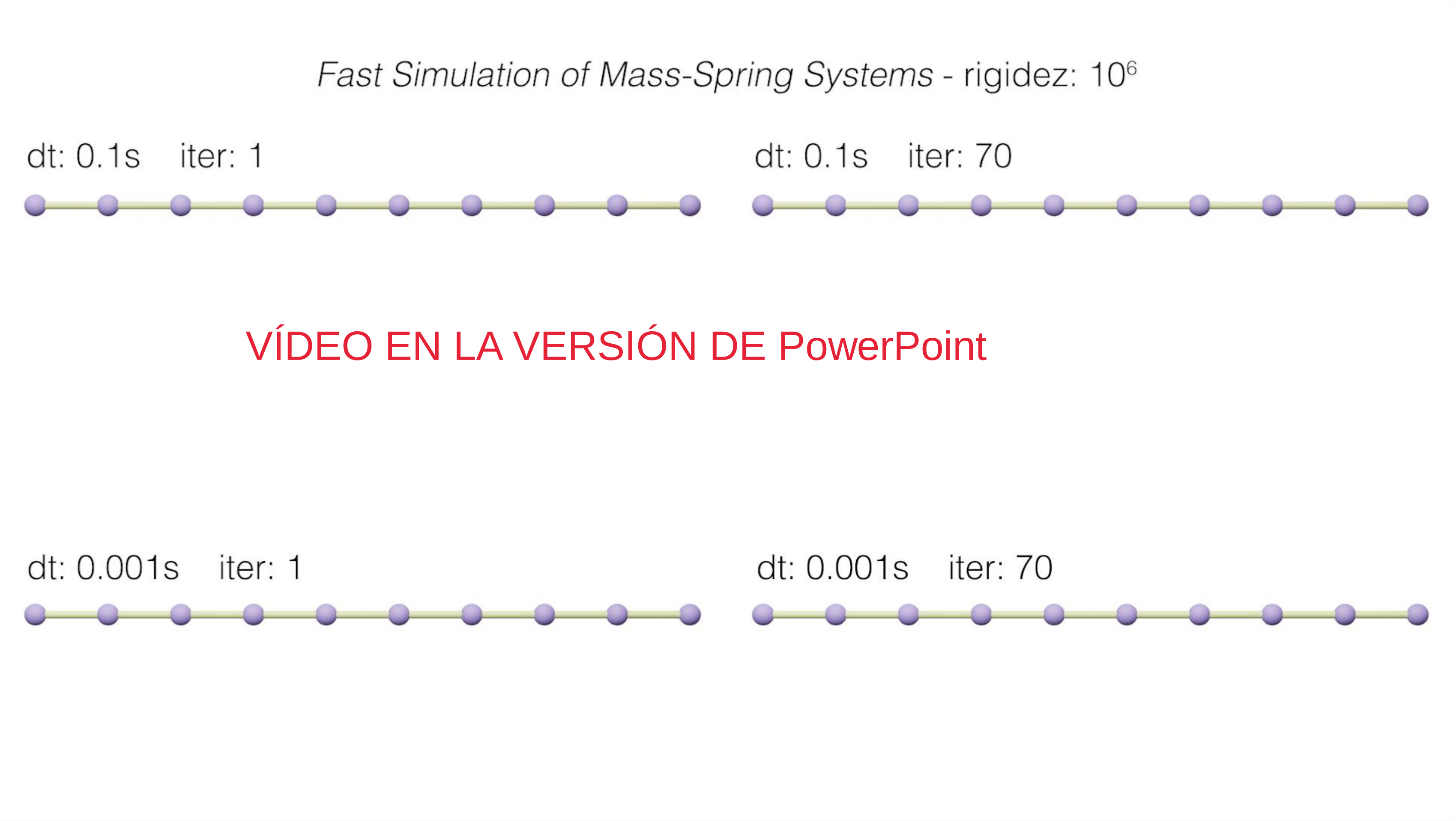
5.2 Estudiando la estática



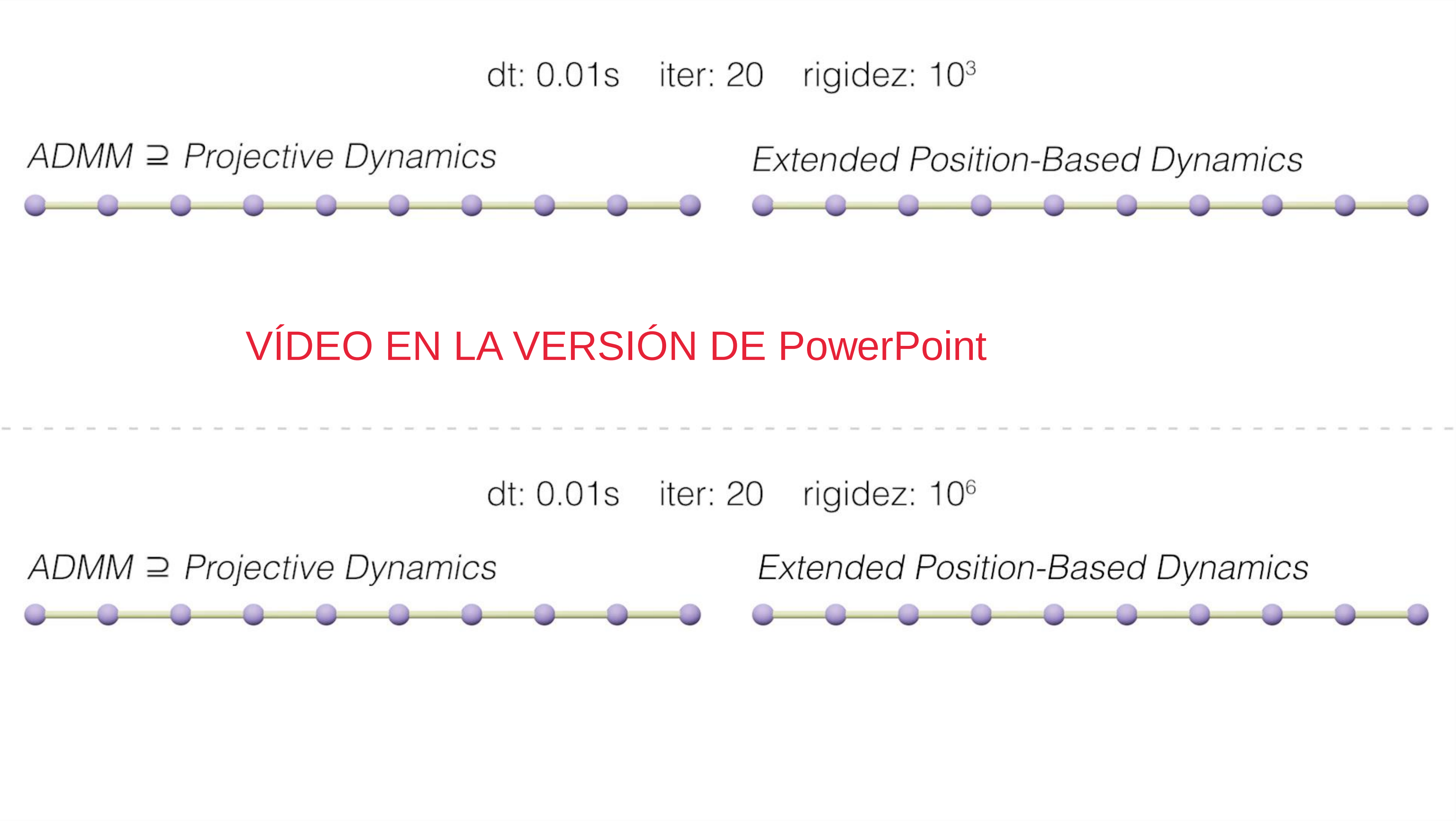
5.2 Estudiando la estática

- En general: $\Uparrow$ trabajo del sistema $\Downarrow$ error
- XPBD: no es tan preciso: $\Downarrow$ iteraciones puede no convergir ✗
- Projective Dynamics: consigue mayor precisión ✓
  - Fast Simulation of Mass-Spring Systems: limitado ✗
  - ADMM $\supseteq$ Projective Dynamics: ✓

5.3 Estudiando la dinámica



5.3 Estudiando la dinámica



VÍDEO EN LA VERSIÓN DE PowerPoint

5.3 Estudiando la dinámica

- En general:
  - El cambio del *dt* influye en el comportamiento
  - $\Uparrow$ rigidez $\Downarrow$ comportamiento correcto
- Fast Simulation of Mass-Spring Systems: $\Downarrow$ iteraciones $\Uparrow$ amortiguamiento ✗
- **ADMM $\supseteq$ Projective Dynamics:**
- **XPBD:**

visualmente plausible con iteraciones medias ✓

6. Conclusiones

6. Conclusiones

- Hemos estudiado e implementado distintos *solvers* de dos familias actuales
    - \+ Backward Euler Implícito
- Hemos definido las condiciones de contorno
- Hemos desarrollado una batería de pruebas para compararlos

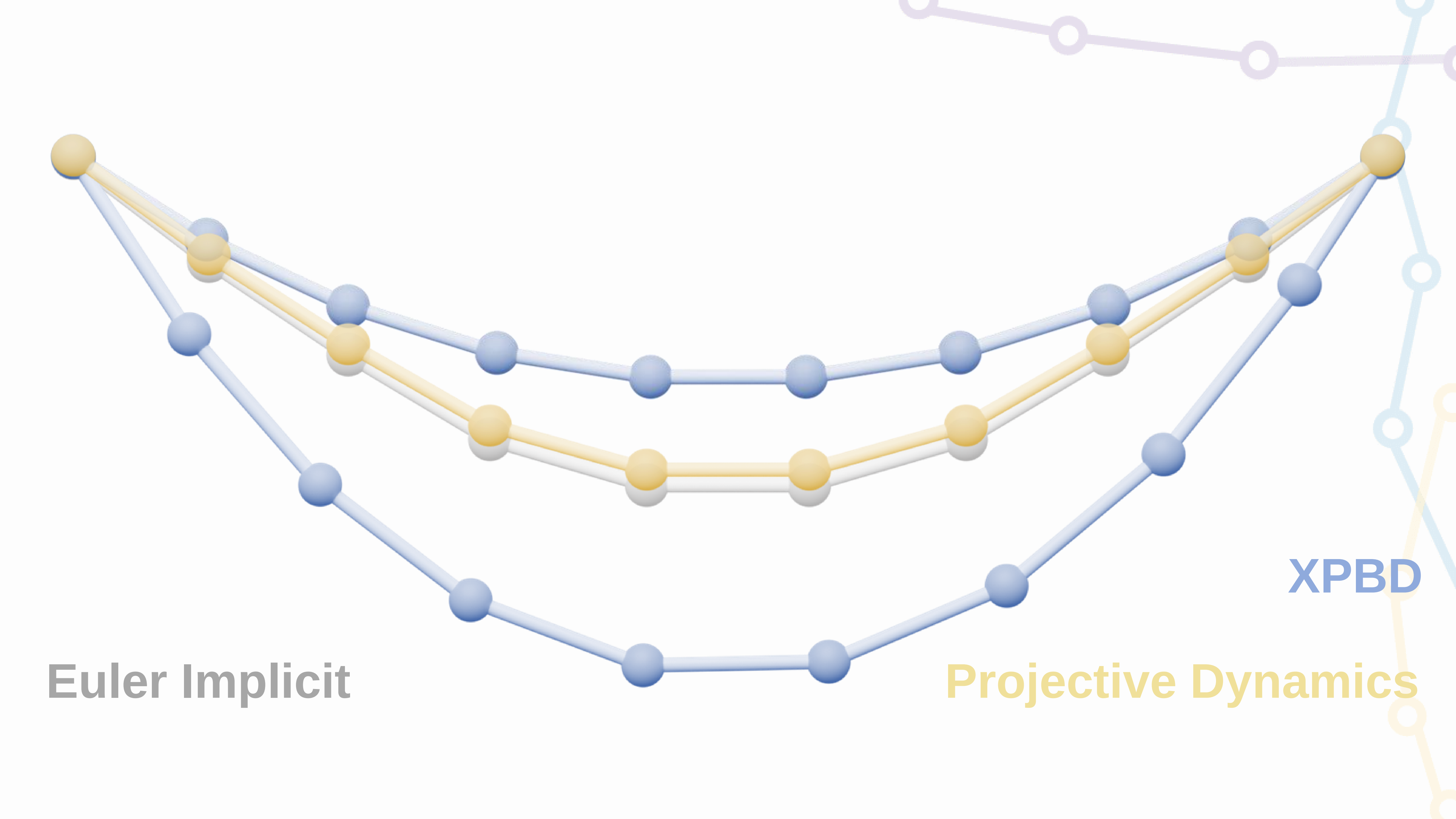
6. Conclusiones

- XPBD: obtiene un buen comportamiento dinámico pero presenta mayor error
- Fast simulation of Mass-Spring Systems: en ciertas situaciones introduce demasiada amortiguación. Al derivar directamente de Euler Implícito, alcanza con poco error la solución.
- ADMM ⊇ Projective Dynamics: en general, reúne las dos características positivas de los anteriores métodos.

6. Conclusiones

- ADMM obtiene resultados bastante positivos
- Trabajo futuro:
  - Aumentar dimensionalidades del problema
  - Continuar el estudio de métodos basados en Projective Dynamics

Gracias por su atención


XPBD
Euler Implicit
Projective Dynamics