

# Layout de Grafos en 3D con Modelos de Fuerzas

DE LA ENERGÍA A LA DEMO INTERACTIVA

---

Sebastián Alzate

11 de diciembre de 2025

COMP6838 — Visualización de Grafos

## ¿Por qué importa el layout en 3D?

- Los grafos reales son densos: necesitamos **estructura visible**.
- 3D ofrece **más libertad**: menos cruces y mejor separación.
- Casos de uso: comunidades, *hubs*, *puentes/bottlenecks*.

### Mensaje

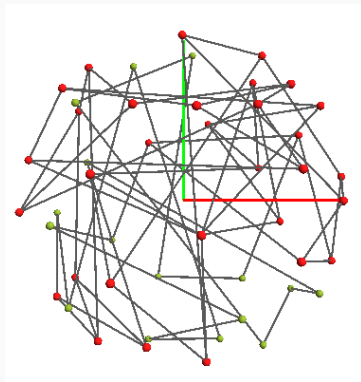
El layout no es decoración: es un **mapeo físico** que sugiere hipótesis sobre la estructura.

## ¿Qué problema resolvemos?

- **Input:** grafo no dirigido  $G = (V, E)$ ,  $|V| = N$ .
- **Output:** posiciones  $P = \{p_i \in \mathbb{R}^3\}$  con:
  1. repulsión global y atracción en aristas,
  2. sin solapamientos,
  3. **estables** para interacción (rotar, hacer zoom, ajustar parámetros).

- **Interactividad:** 30–60 FPS en grafos medianos.
- **Robustez:** datos ruidosos, aristas duplicadas, lazos.
- **Control visual:** pocos parámetros, interpretables.
- **Lectura:** evitar *hairballs*; resaltar módulos y puentes.

- **Modelos:** FR, LinLog y Spring-Electrical (SE) conmutables.
- **Termodinámica:** temperatura  $T$ , *cooling* y *reheat*.
- **Estabilidad:** *clamping*, *recenter*, resorte radial.
- **Suavizado:** EMA con factor  $\alpha$  para eliminar *jitter*.
- **Comunidades:** k-means sobre posiciones para colorear.



## Fundamento físico (modelo de partículas)

- Cada nodo  $i$  es una partícula en  $\mathbb{R}^3$  con posición  $p_i$ .
- **Repulsión** entre todos los pares; **atracción** en las aristas.
- El layout minimiza una **energía potencial**  $E(p_1, \dots, p_N)$ .
- La simulación es un **descenso de energía** con paso regulado por  $T$ .

**Energía total.** Cada modelo define una energía  $E(P)$  cuyas derivadas dan las fuerzas:

$$E(P) = \begin{cases} \sum_{i < j} k^2 \ln d_{ij} + \sum_{(i,j) \in E} \frac{1}{3k} d_{ij}^3 & \text{(FR)} \\ \sum_{(i,j) \in E} w_{ij} \ln d_{ij} + \alpha \sum_{i < j} \ln d_{ij} & \text{(LinLog)} \\ \sum_{(i,j) \in E} \frac{C_a}{2} (d_{ij} - L)^2 + \sum_{i < j} \frac{C_r}{d_{ij}} & \text{(SE)} \end{cases}$$

con  $d_{ij} = \|p_i - p_j\|$ . La dinámica implementa un **descenso de energía**:

$$p_i \leftarrow p_i + \min\left(1, \frac{T}{\|D_i\|}\right) D_i, \quad D_i = -\nabla_{p_i} E(P).$$

## Por qué converge en la práctica.

- Para pasos suficientemente pequeños,  $E(P)$  es *función de Lyapunov*: decrece monótonamente.
- El *cooling* reduce el paso efectivo; el *reheat* evita quedar atrapado en malos mínimos locales.
- *Invariancias*: traslación/rotación/escala no cambian  $E$ ; por eso usamos **recentering** y **clamping**.
- *No convexidad*: hay múltiples mínimos; la inicialización y el modelo seleccionan la “narrativa” del layout.



# Por qué estos layouts muestran estructura (en una frase por idea)

## Comunidades visibles.

- **LinLog**: la atracción crece muy lento con la distancia  $\log(1 + d)$ , así que no “obliga” a pegar bloques lejanos; quedarse en módulos cercanos baja la energía. Resultado: clusters separados y limpios.

## Hubs centrados.

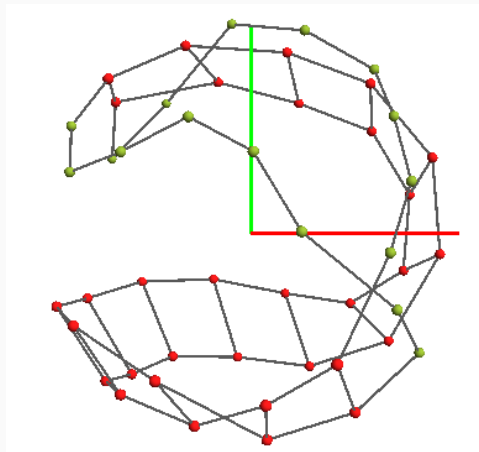
- **FR**: fija una escala típica  $k \propto N^{-1/3}$ ; un hub siente muchas atracciones en varias direcciones y se estabiliza cerca del centro geométrico del grupo.

## Control fino (SE).

- **Spring–Electrical**: la repulsión  $1/d^2$  evita colapsos;  $L$  marca la longitud “ideal” por arista y  $C_a$  su rigidez. Ajustando  $(C_r, C_a, L)$  abrimos o compactamos el dibujo sin cambiar la topología.

## Modelo 1: Fruchterman–Reingold (FR)

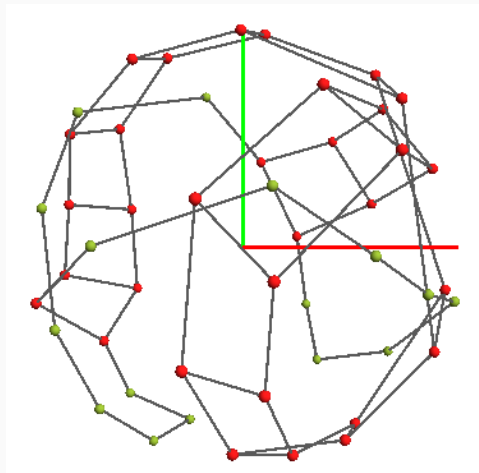
- Repulsión:  $F_{\text{rep}} \propto \frac{k^2}{d}$ .
- Atracción:  $F_{\text{att}} \propto \frac{d^2}{k}$ .
- $k \approx 2/\sqrt[3]{N}$  como escala típica.
- **Efecto:** layouts compactos y limpios; buen *baseline*.



Karate con FR

## Modelo 2: LinLog

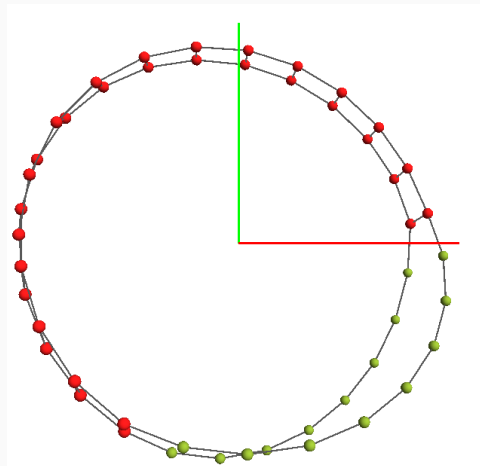
- Repulsión suave (similar a FR).
- Atracción sublineal: efecto logarítmico en la distancia.
- **Efecto:** separa mejor comunidades; realza *puentes*.



Karate con LinLog

## Modelo 3: Spring-Electrical (SE)

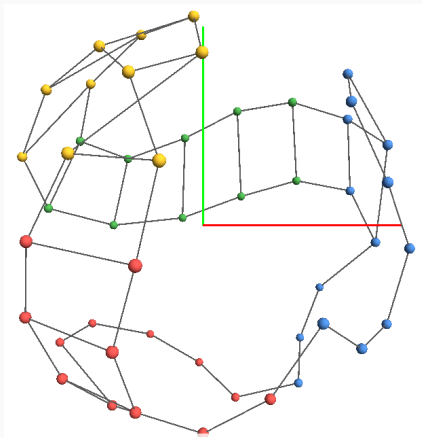
- Repulsión Coulomb:  $F_{\text{rep}} \propto \frac{C_r}{d^2}$ .
- Resorte lineal:  $F_{\text{att}} = C_a (d - L)$ .
- Parámetros interpretables:  $C_r$  (apertura),  $C_a$  (rigidez),  $L$  (longitud ideal).



Karate con SE

## Coloración por comunidades (k-means++)

- K-means con inicialización *k-means++* sobre posiciones (suavizadas).
- Etiquetas estables y paleta discreta de 12 colores.
- Lectura: bloques densos, *hubs* y *brokers* visibles.



- **Contexto:** red social de un dojo universitario (Zachary, 1977). Dos líderes: *Mr Hi* y *John A*.
- **Nodos:** miembros del club (*Mr Hi*, *John A*, *Actor 2*, ...). Tamaño típico:  $N \approx 34\text{--}35$ .
- **Aristas:** lazos de interacción/amistad observados; grafo **no dirigido**, **no ponderado**.

## ¿Por qué aplicar un layout por fuerzas aquí?

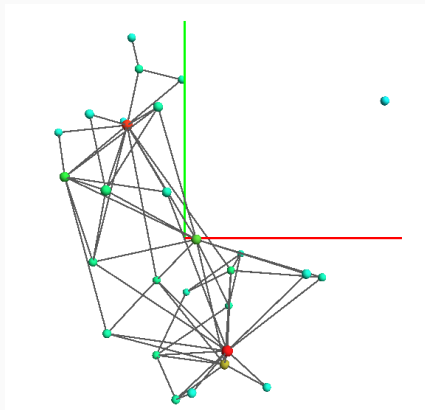
- **Hipótesis visual:** las dos facciones deberían separarse en el espacio embebido.
- **Brooks/puentes:** nodos que conectan con ambos líderes deben *quedar en el borde* entre módulos.
- **Interactividad:** los modelos por fuerzas permiten ajustar temperatura y parámetros para estabilizar la lectura en vivo.

### Qué buscamos ver

- Facciones separadas y densas.
- *Hubs* alrededor de cada líder.
- Puentes destacados en el “valle” entre grupos.

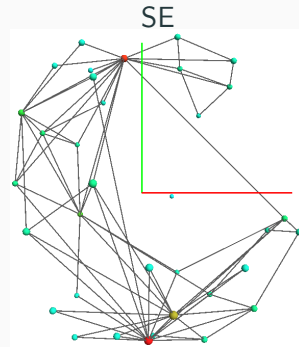
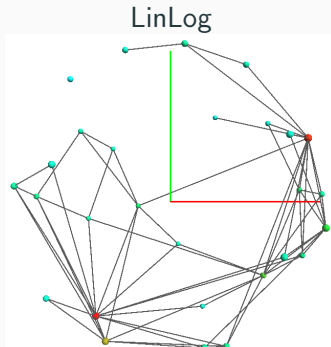
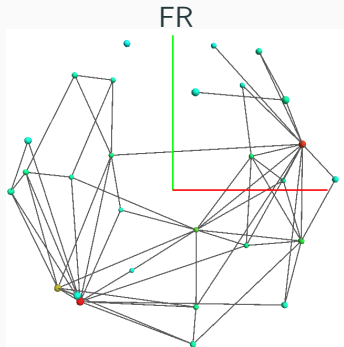
## Caso de estudio: Karate Club de Zachary

- Grafo social clásico ( $N \approx 34-35$ ,  $|E| \approx 78$ ).
- Dos facciones emergen y unos pocos puentes las conectan.
- Objetivo: resaltar módulos y localizar *hubs/brokers*.





# Comparación rápida de modelos (mismo grafo)



## Conclusiones del ejemplo (Karate)

- **Separación de facciones:** LinLog y SE muestran con mayor claridad los dos bloques; FR da una vista base compacta y estable.
- **Hubs:** *Mr Hi* y *John A* aparecen como centros locales de alta conectividad.
- **Ajustes útiles:**
  - SE con  $C_r \uparrow$  abre el núcleo y hace visibles los puentes.
  - $L$  en SE controla la longitud típica de aristas; subirlo separa módulos sin romperlos.
  - Suavizado (EMA) mejora la lectura durante la demo sin afectar la física.
- **Resumen operativo:** FR para baseline, LinLog para resaltar comunidades, SE para afinar el espaciado y presentar.

# Ventajas y desventajas

Modelo	Ventajas	Desventajas
FR	Compacto, estable, pocos parámetros.	Puede ocultar fronteras comunitarias; menos control de espaciado.
LinLog	Separa módulos y muestra puentes con claridad.	Puede sobredispersar periferias si no se calibra.
SE	Parámetros interpretables ( $C_r$ , $C_a$ , $L$ ); <i>tuning</i> fino.	Más sensible a calibración; riesgo de layouts “forzados”.

# Controles prácticos y cuándo tocarlos

## Termodinámica

- $T$ : subir si todo está “pegado”; bajar si vibra.
- *Cooling on/off*: pausar enfriamiento tras cambios grandes.
- *Reheat*: algunas iteraciones calientes para reacomodar.

## Suavizado

- EMA  $\alpha$ : alto para presentaciones; bajo para explorar.

## SE (afinación fina)

- $C_r$ : aumenta apertura del núcleo.
- $C_a$ : controla rigidez de los resortes.
- $L$ : sube para abrir módulos, baja para compactar.