



**UNIVERSIDAD NACIONAL DEL LITORAL**  
FACULTAD DE INGENIERÍA Y CIENCIAS HÍDRICAS

Tecnicatura en diseño y programación de videojuegos

# Modelos y Algoritmos para Videojuegos II

## Unidad 3 La vida secreta de los resortes en Box2D

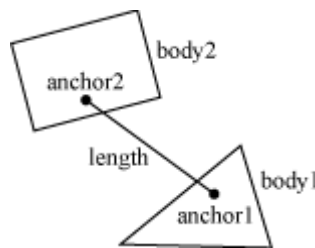
Docente  
Lucas Sebastián Villa

## La vida secreta de los resortes en Box2D

Para simular resortes en Box2D necesitamos recurrir a los famosos Joints. Un Joint es una restricción que se establece sobre dos cuerpos dinámicos de Box2D. Las restricciones pueden ser de distinto tipo. Este es un tema que veremos más adelante en profundidad. De momento sólo haremos una introducción a cómo simular resortes.

### DistanceJoint

El Joint que utilizaremos para simular resortes es denominado DistanceJoint.



En su forma más sencilla este Joint no se comporta como un resorte, sino que simplemente mantiene la distancia entre dos cuerpos. Para lograr que se comporte como un resorte debemos modificar un poco los parámetros que lo definen. Para ello veamos primero cuáles son los parámetros necesarios para crearlo. Para crear un DistanceJoint primero debemos crear un objeto DistanceJointDef que contiene una descripción de la restricción:

```
b2DistanceJointDef jointDef;
```

Luego debemos inicializar el objeto:

```
jointDef.Initialize(b2Body* first, b2Body* second, b2Vec2 anchorFirst, b2Vec2 anchorSecond);
```

En dicha inicialización los objetos first y second son punteros a los dos objetos que se encuentran unidos por el resorte. Luego debemos pasarle los puntos de anclaje de cada objeto. Estos puntos como se muestra en el dibujo indican en qué parte de los mismos está prendido el resorte. Estos puntos deben ser dados en coordenadas del mundo.

Luego de haber inicializado la restricción podemos proseguir a configurarla como un resorte. En particular hay 3 propiedades con las cuales debemos jugar para lograr el comportamiento elástico: length, frequencyHz y damp.

## Length

Esta propiedad indica cuál es la distancia de equilibrio del resorte.

## FrequencyHz

Esta propiedad indica la frecuencia a la cual oscilará el resorte. En Box2D no utilizamos la constante  $K$  del resorte, sino que utilizamos este parámetro que a su vez sí depende de la constante  $K$ . La frecuencia de oscilación se calcula como:

$$\omega_0 = \sqrt{\frac{K}{m}}$$

Dónde  $\omega_0$  es la frecuencia,  $K$  es la constante del resorte y  $m$  es la masa a la cual se encuentra unido el resorte.

## Damp

La propiedad `damp` establece la atenuación del sistema. Es decir, mide que tanto se atenuarán las oscilaciones. Si el `damp` es igual a 0 el sistema no debería detenerse nunca idealmente.

Veamos como configuramos entonces en código un resorte usando la definición del joint inicializada anteriormente:

```
jointDef.length=1.0; jointDef.frequencyHz=10; jointDef.dampingRatio=0.5;  
y finalmente creamos el objeto join:
```

```
b2DistanceJoint *distJoint=  
(b2DistanceJoint*)phyWorld->CreateJoint(&jointDef);
```

Dónde le pasamos como argumento la definición creada anteriormente.

## Observaciones

Simular los resortes de esta forma es muy sencillo, pero no es muy exacto físicamente ya que para lograr estabilidad los resortes son sobre amortiguados incluso cuando indicamos que no deseamos *damp*. Esto es muy común en los motores físicos. Como ejercicio puede implementar un resorte a mano usando dos cuerpos de Box2D y calculando la fuerza como dicen los apuntes de teoría y comparar los resultados.