

INGENIERÍA INFORMÁTICA - ITBA

PROYECTO FINAL

***SIMULACIÓN Y ANIMACIÓN BIOMECÁNICA
DE UN HUMANOIDE***

Alumnos:

Altamiranda Graterol, Enzo

Fontanella De Santis, Teresa

Mehdi, Tomás

Tutor:

Dr. Daniel Ricardo Parisi

Año 2016

Agenda

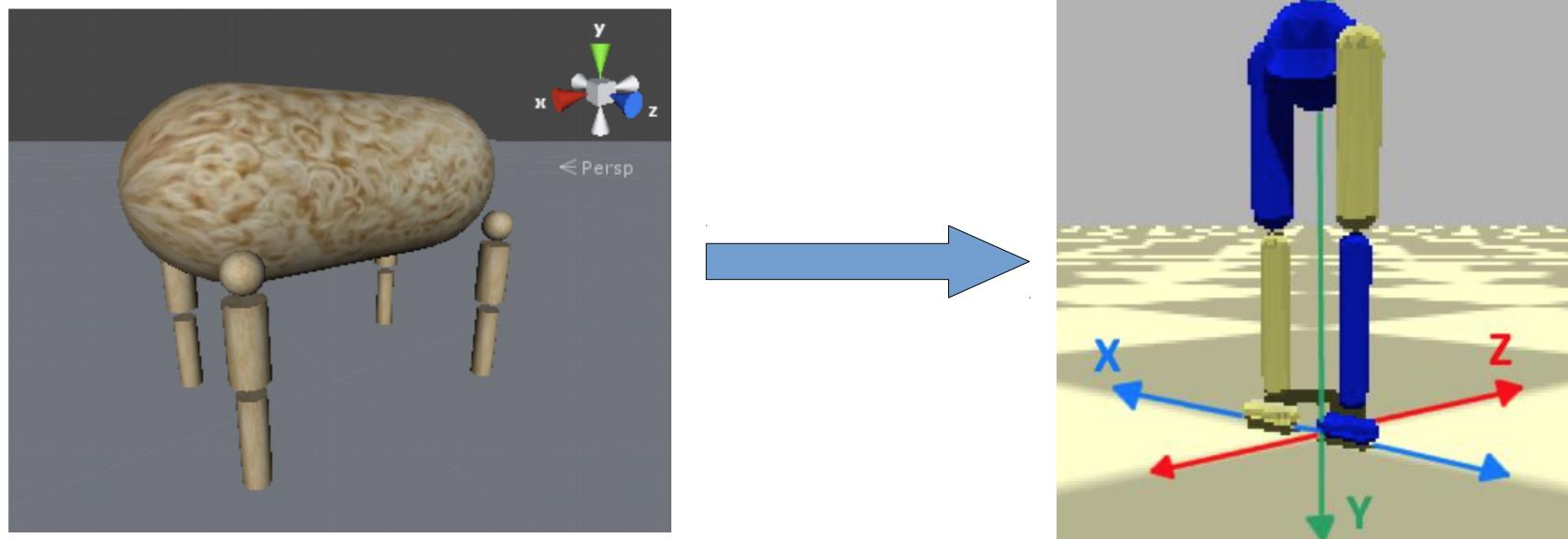
Introducción

El objetivo es lograr la simulación dinámica de la caminata de un humanoide.

- Simulación
- Dinámica
- Humanoide

Introducción

De cuadrúpedos a bípedos.



Técnicas de simulación



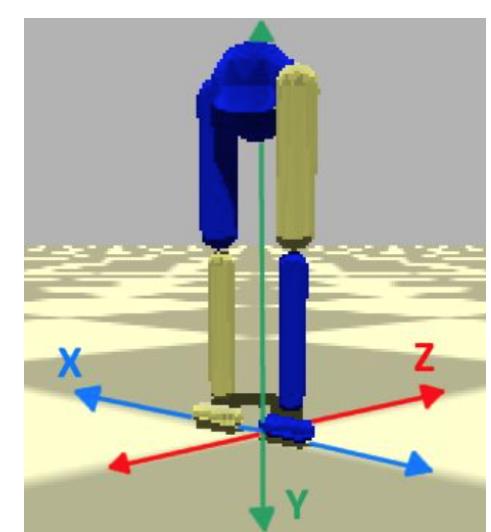
Simulación Cinemática
(método manual)



Passive Walker



Métodos Basados en
Control



Métodos Basados en
Búsqueda

Requerimientos

1) Motor Físico:

Realista vs Cuasi Realista.

2) Optimización de Parámetros:

Algoritmo de Optimización.

3) Modelo del Humanoide:

A construir por medio de software.

Herramientas utilizadas

1) Motor Físico:

Bullet Physics en C++

2) Optimización de Parámetros:

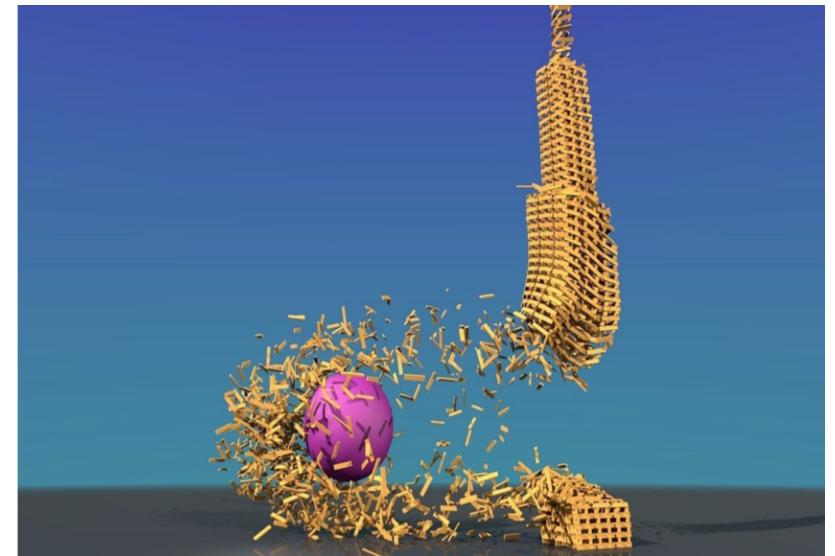
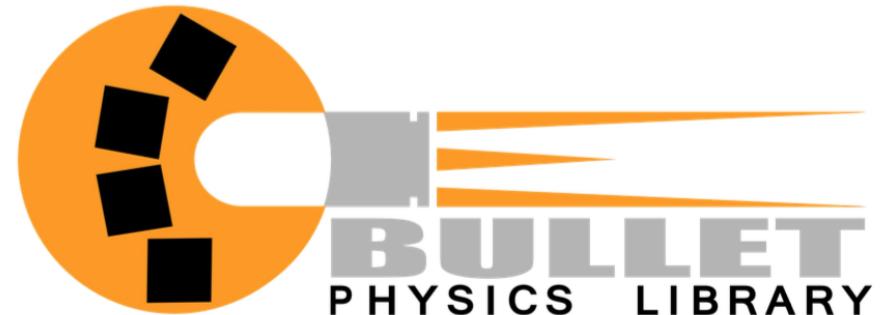
Algoritmos Genéticos

3) Modelo del Humanoide:

Jerarquía de clases escrita en C++

Motor Físico - *Bullet Physics*

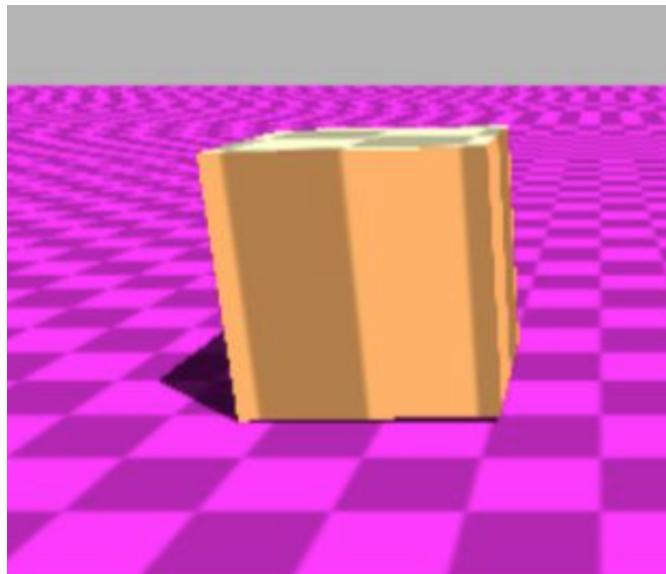
- Motor de simulación física realista.
- Hecho en C++.
- Trae una interfaz gráfica simple.
- Simulación por medio de resolución de sistemas de ecuaciones.



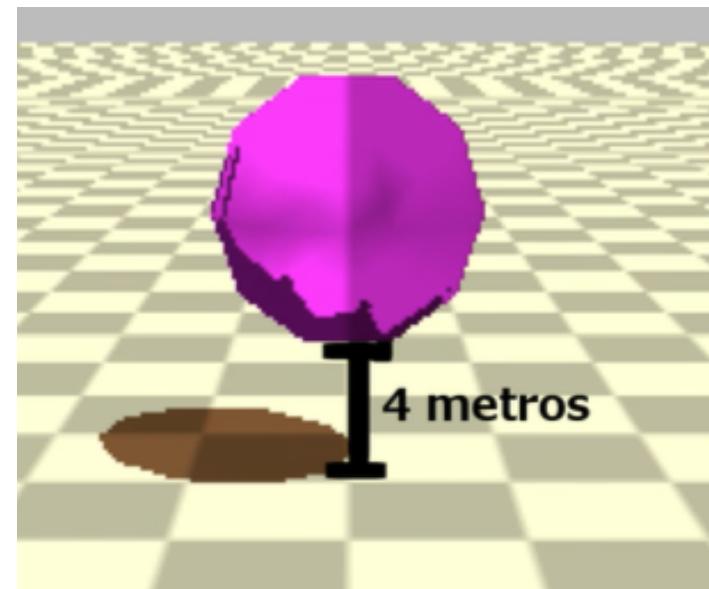
Motor Físico - *Experimentos*

Se realizaron experimentos para analizar la afinidad del simulador a ecuaciones físicas de dinámica. En particular se estudió la **fricción** y elasticidad de las **colisiones**.

Fricción con el piso



Colisión



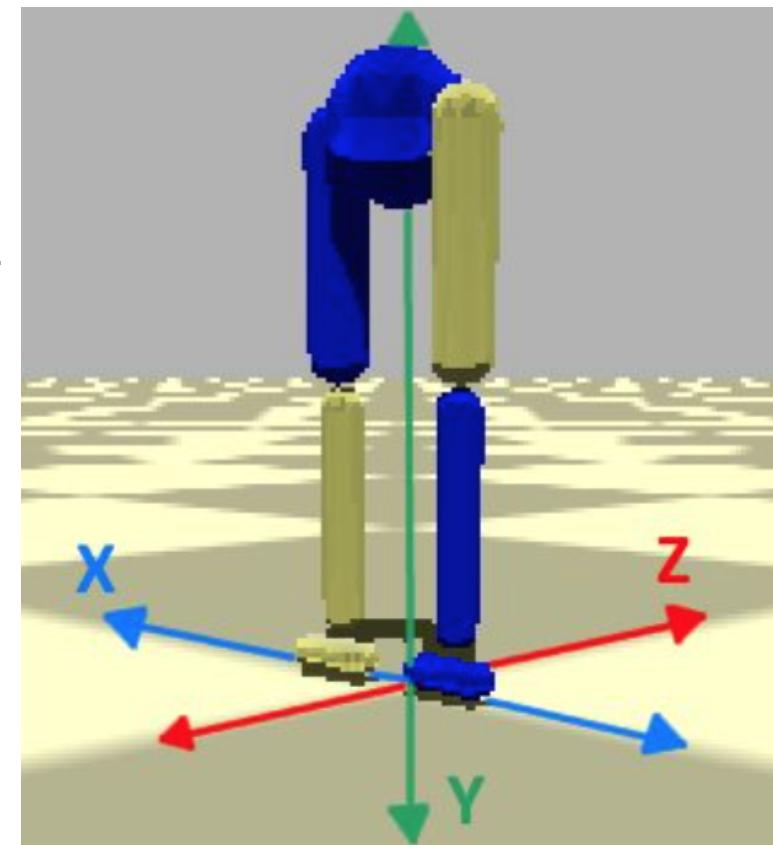
Optimizador - *Algoritmo genético*

Se utilizaron dos librerías implementadas en C++. Se usó la primera y posteriormente se cambió a la segunda.

- 1) Genetic Algorithm Library (Mladen Jankovic)
- 2) Galib (Matthew Wall)

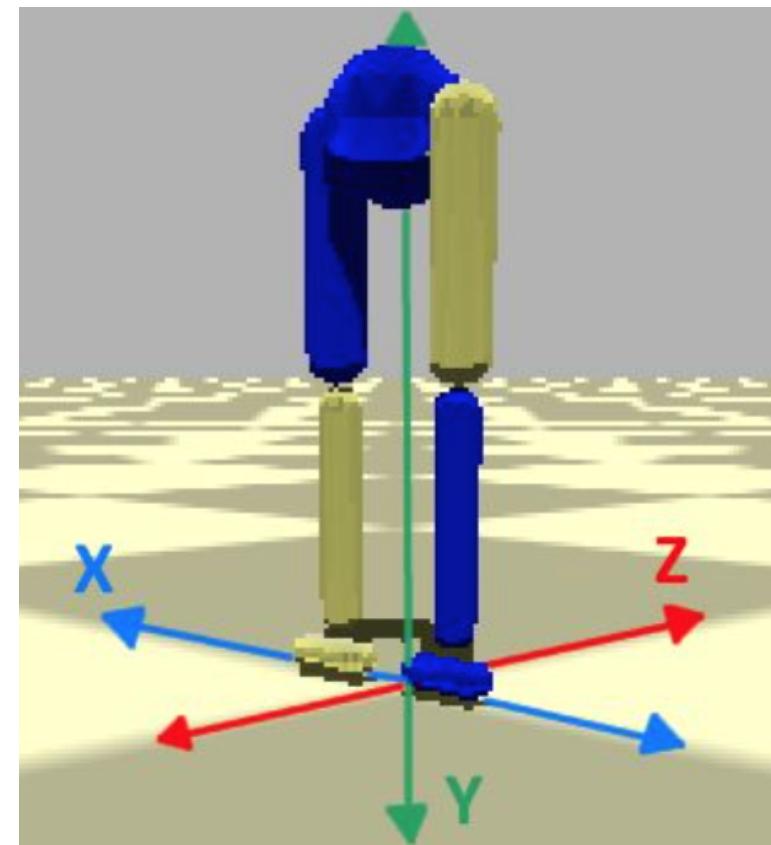
Modelización: Objetos en C++

- El cuerpo se representó como una jerarquía de objetos.
- Cada componente tiene definida una forma de moverse.
- Los componentes están conectados a través de uniones físicas.

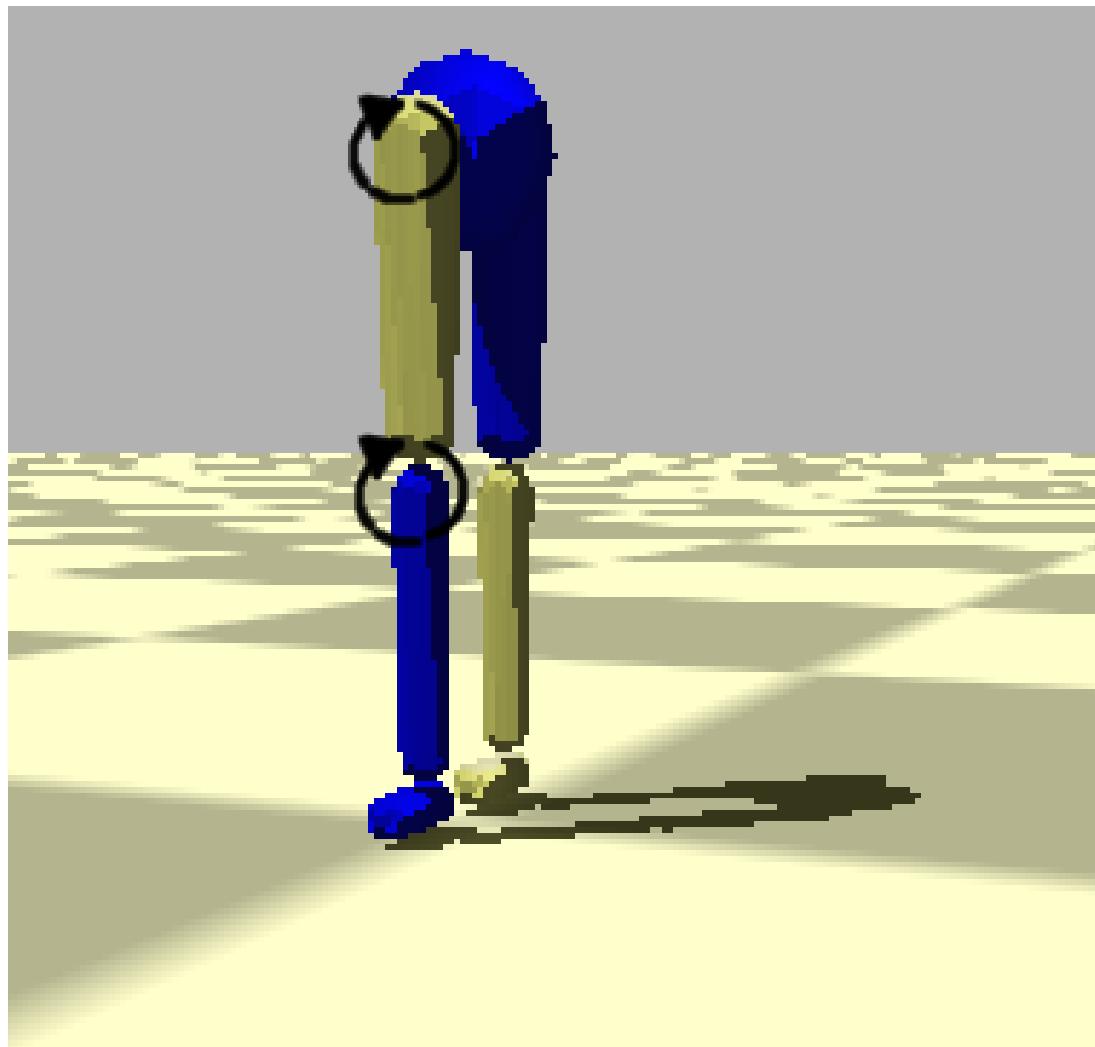


Modelización: Objetos en C++

- Se utilizo solo la parte inferior del cuerpo de un humanoide.
- Se restringieron los grados de libertad de movimiento de muslos, piernas y cadera.

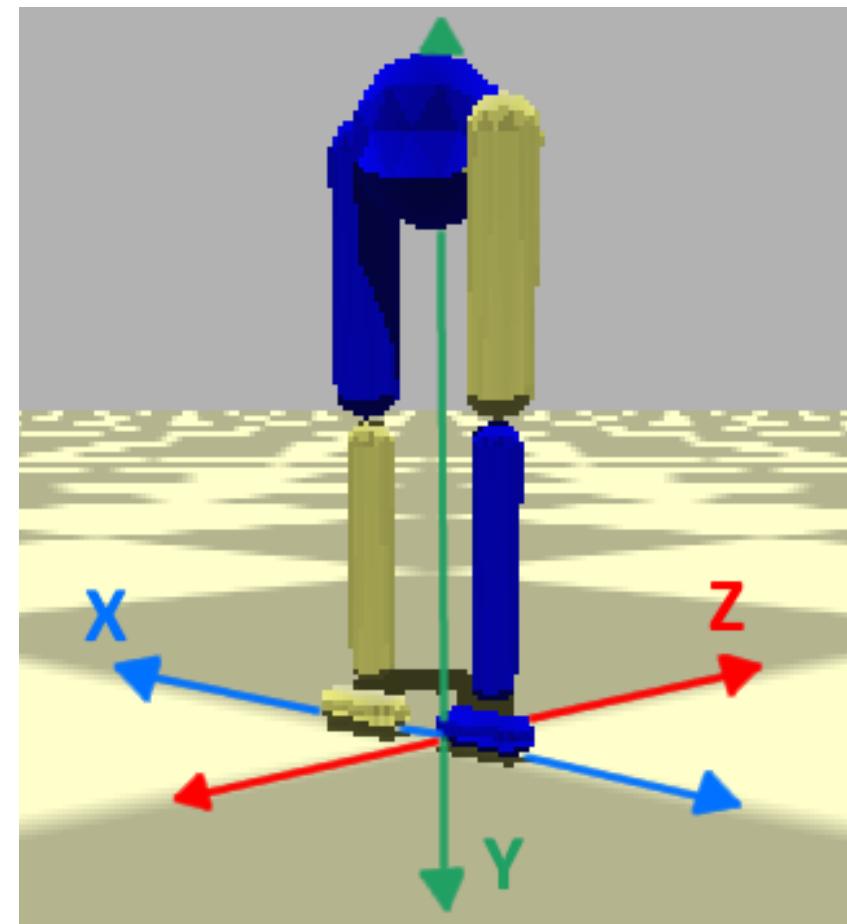


Movimiento de las partes



Movimiento del Humanoide

- Función partida
(primer paso)
- Parte cíclica
(fase sincronizada)



Movimiento del Humanoide

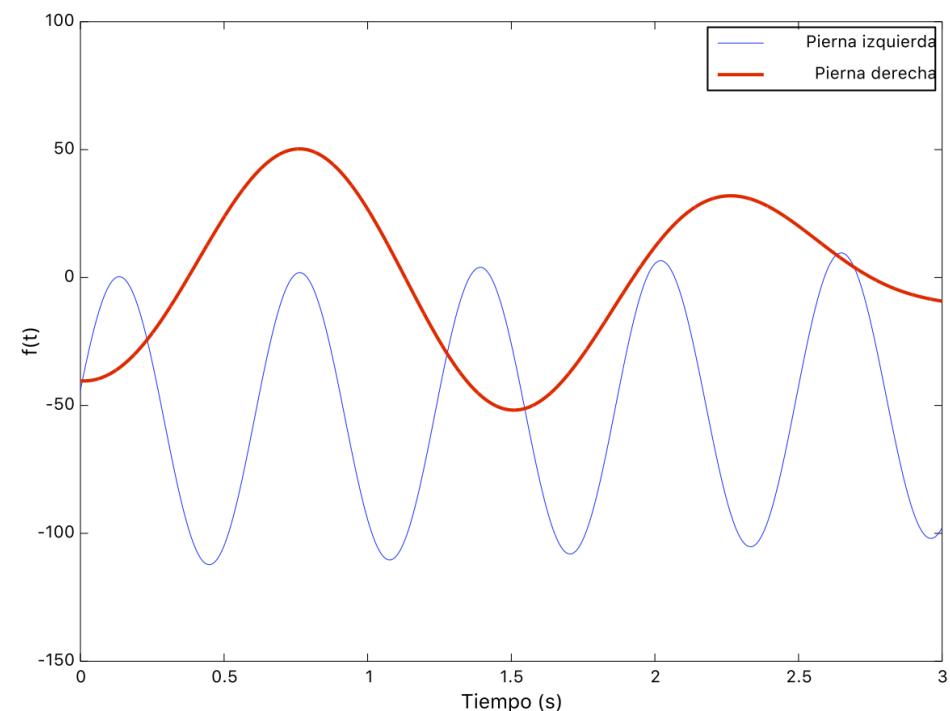


Gráfico del movimiento primer paso

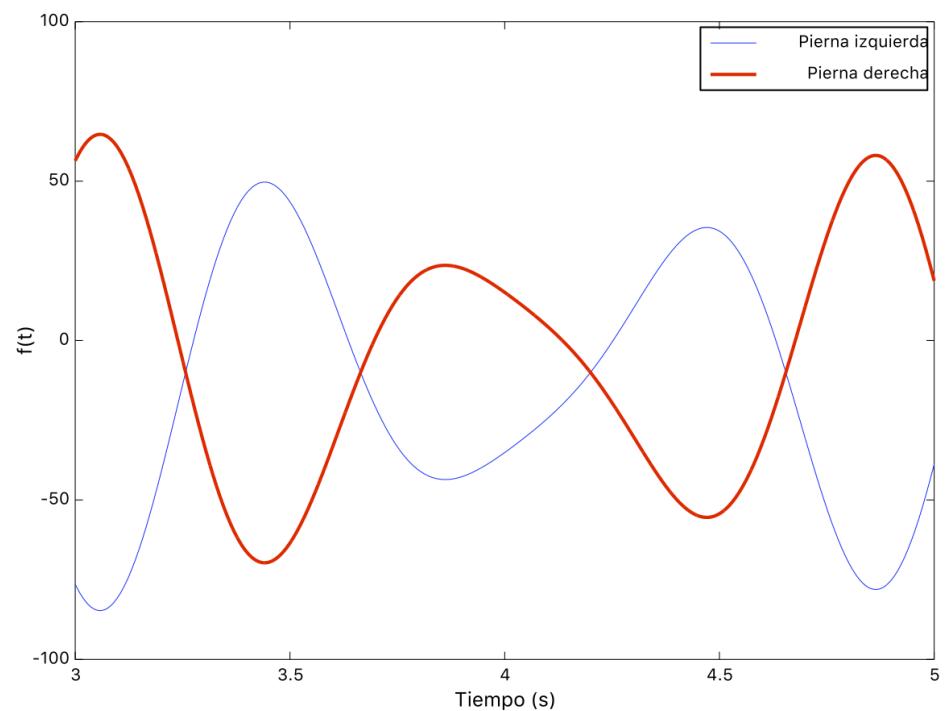


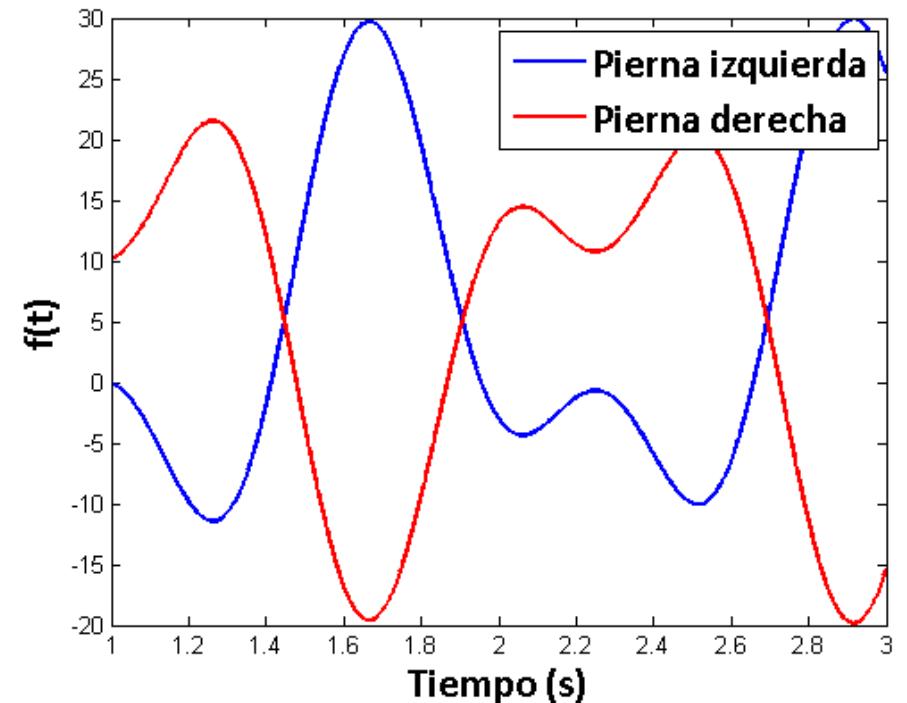
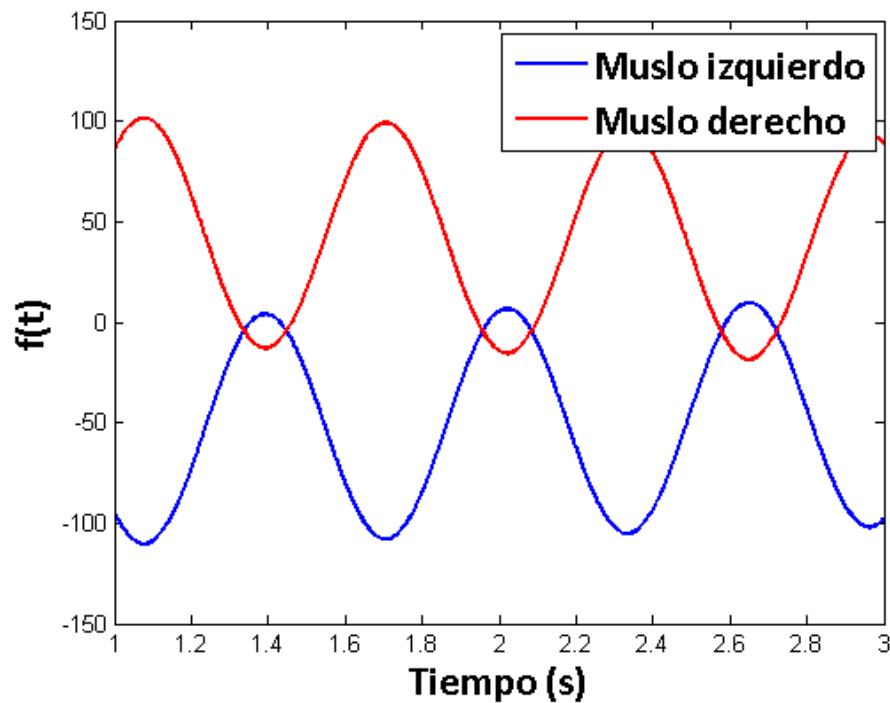
Gráfico del movimiento parte cíclica

Actuadores

- Genérico
- Coseno doble frecuencia
- Fourier de orden 2

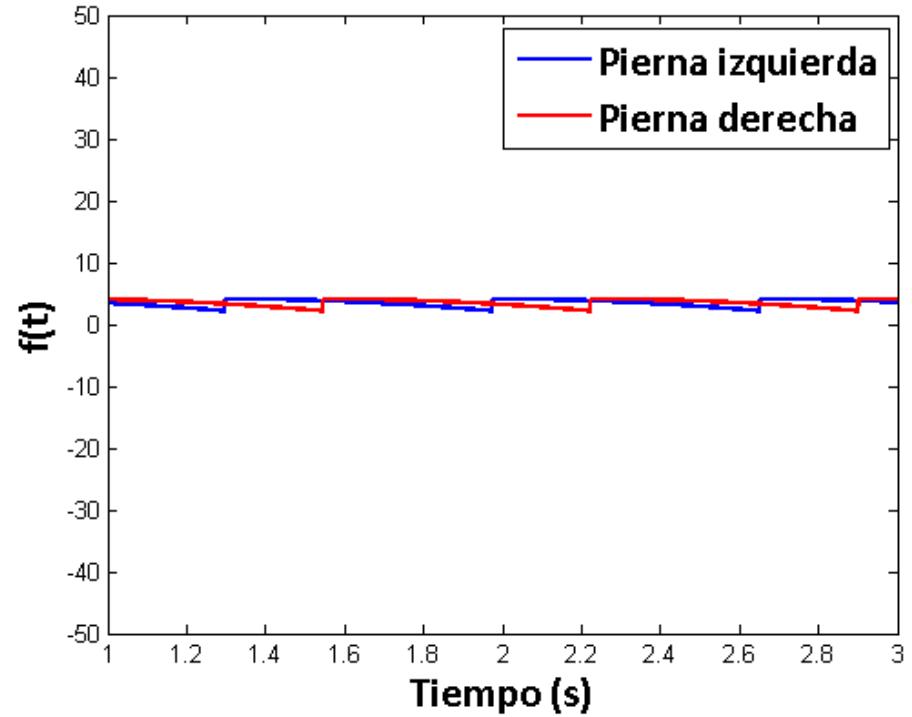
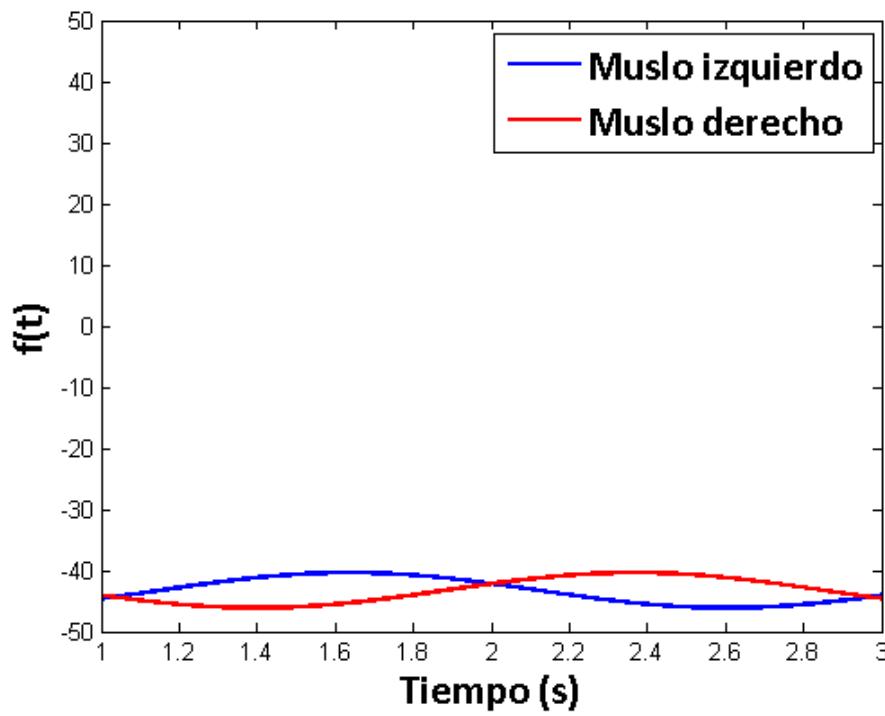
Actuadores - Genérico

$$f(t) = A_1 \sin(\omega_1 t + \phi) + A_2 \cos(\omega_2 t + \phi) + C$$



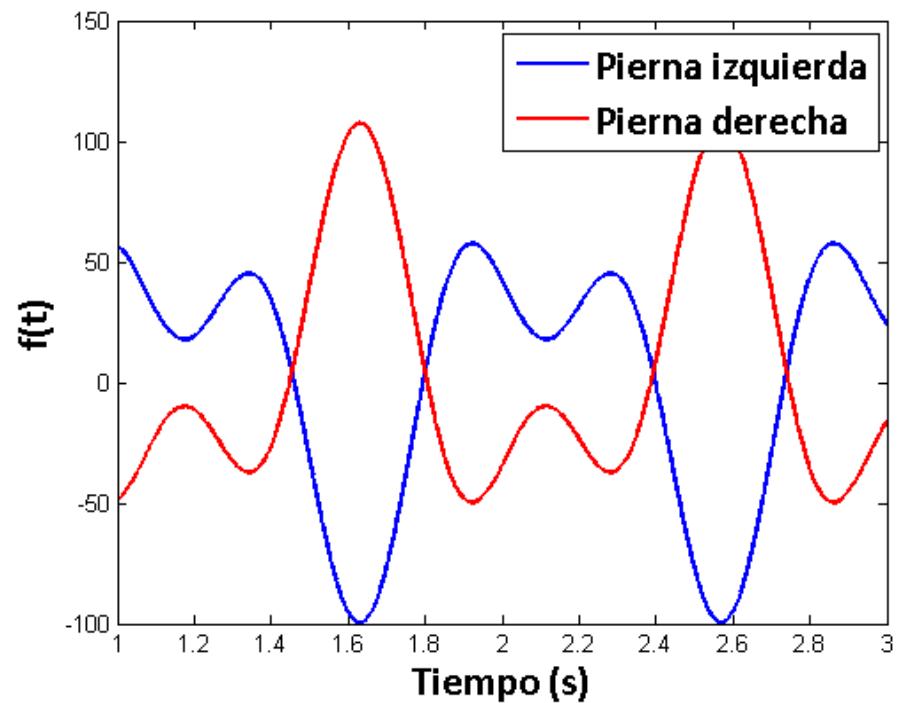
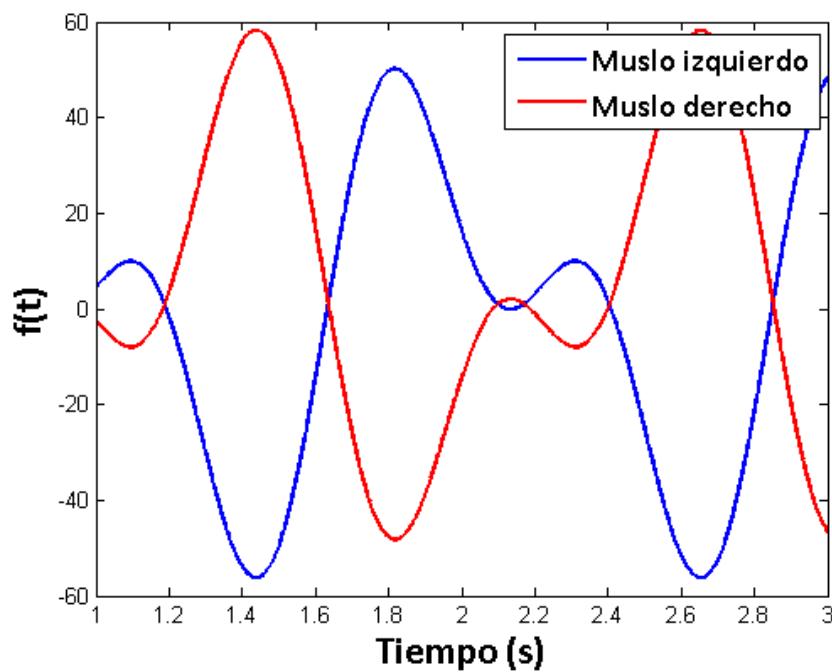
Actuadores - Coseno doble frecuencia

$$f(t) = \begin{cases} A \cos(\omega_1 \psi(t)) + C & \text{si } \omega_1 \psi(t) < \pi \\ A \cos(\omega_2(\psi(t) - (\pi/\omega_1) + (\pi/\omega_2))) + C & \text{en otro caso} \end{cases}$$



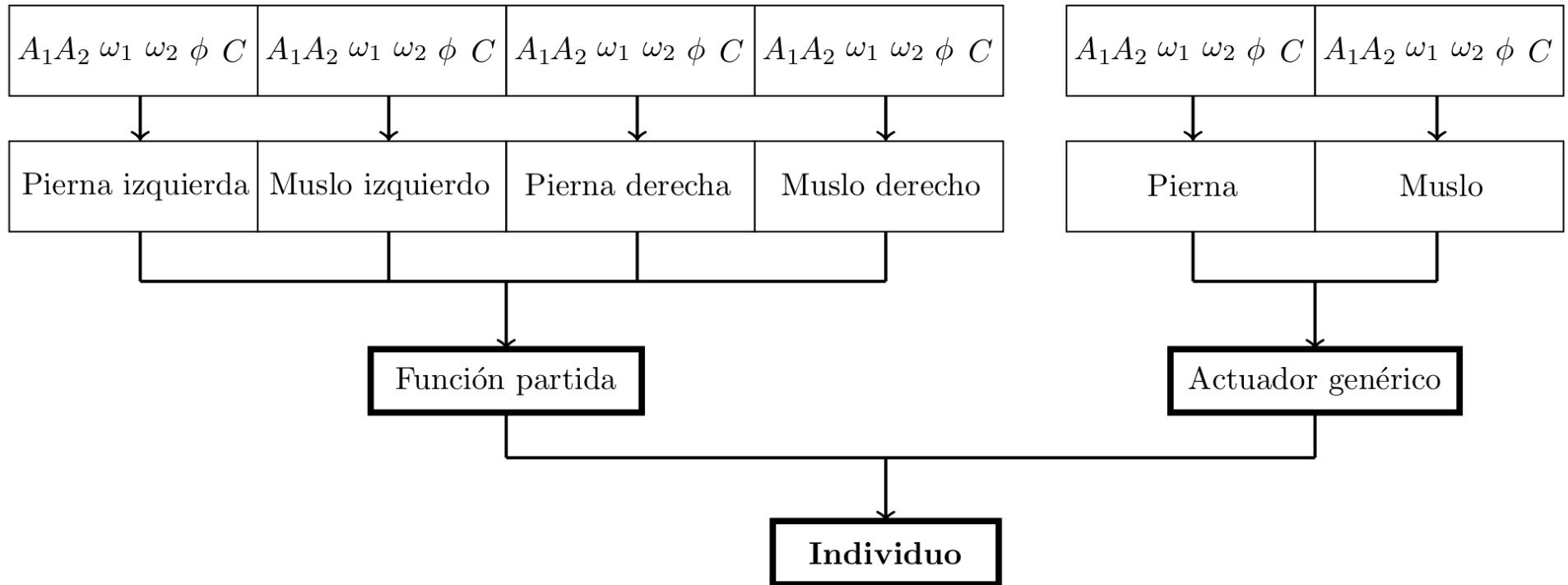
Actuadores - Fourier de orden 2

$$f(t) = A_1 \sin(\omega t + \phi) + B_1 \cos(\omega t + \phi) + A_2 \sin(2\omega t + \phi) + B_2 \cos(2\omega t + \phi) + C$$



Algoritmo genético - *Individuo*

Esquema de un individuo – Ejemplo con función partida y actuador genérico



Algoritmo genético - *Individuo*

Implementación de individuos

Individuo			
	Tipo 1	Tipo 2	Tipo 3
Actuador	Genérico	Genérico	Coseno doble frecuencia
Función partida	No	Sí	Sí
Fase sincronizada	Sí	Sí	Sí

Algoritmo genético - *Individuo*

Constitución del cromosoma

1. Cantidad de parámetros según tipo de actuador y función partida

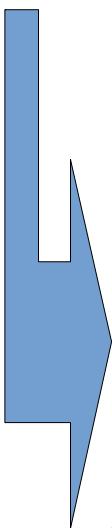
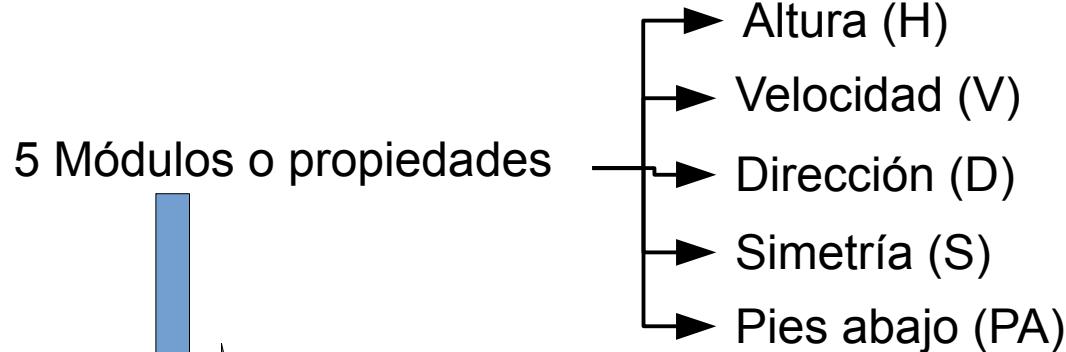
		Parámetro			
		Amplitud	Frecuencia	Fase	Término independiente
Actuador	Genérico	2	2	1	1
	Coseno doble frecuencia	1	2	1	1
Función partida		2	2	1	1

Algoritmo genético - *Individuo*

2. Cantidad de parámetros y longitud del cromosoma según tipo de individuos

Parámetro		Individuo		
		Tipo 1	Tipo 2	Tipo 3
	Amplitud	4	12	8
	Frecuencia	2	6	8
	Fase	2	6	6
	Término independiente	2	6	6
	Totales	10	30	28

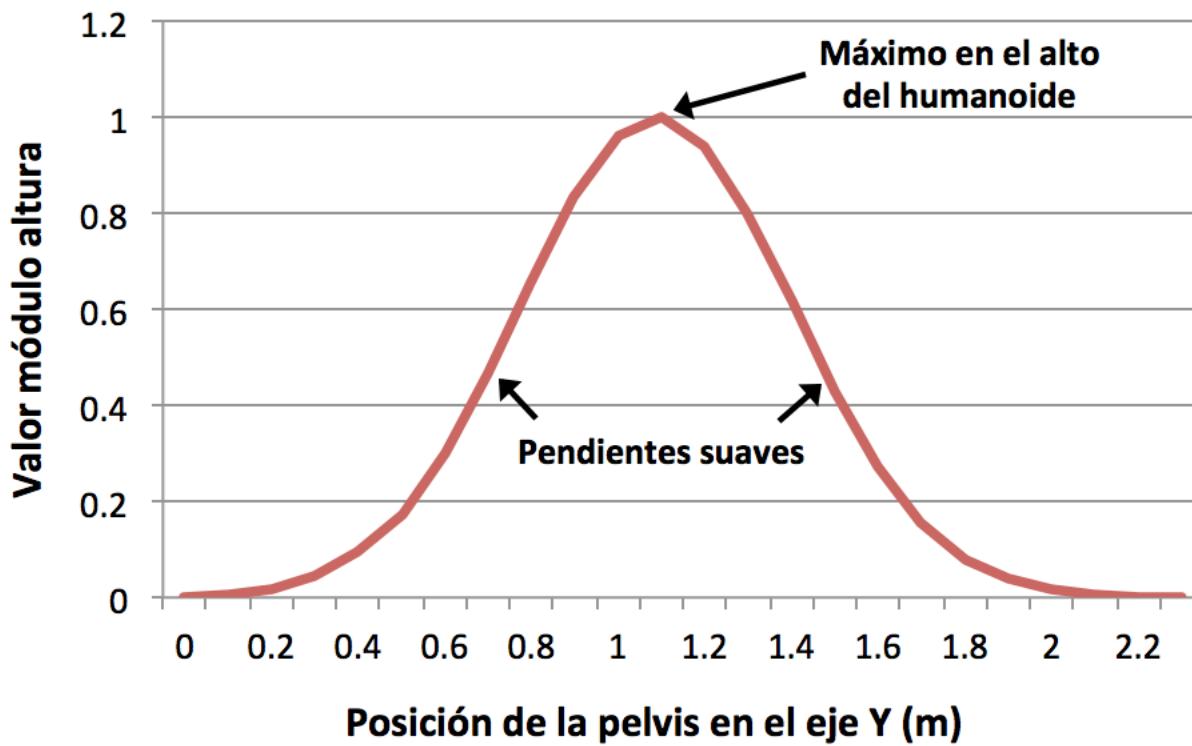
Algoritmo genético - *Fitness*



- Tienen el mismo peso y pueden valer entre 0 y 1
- Están definidos utilizando una función exponencial

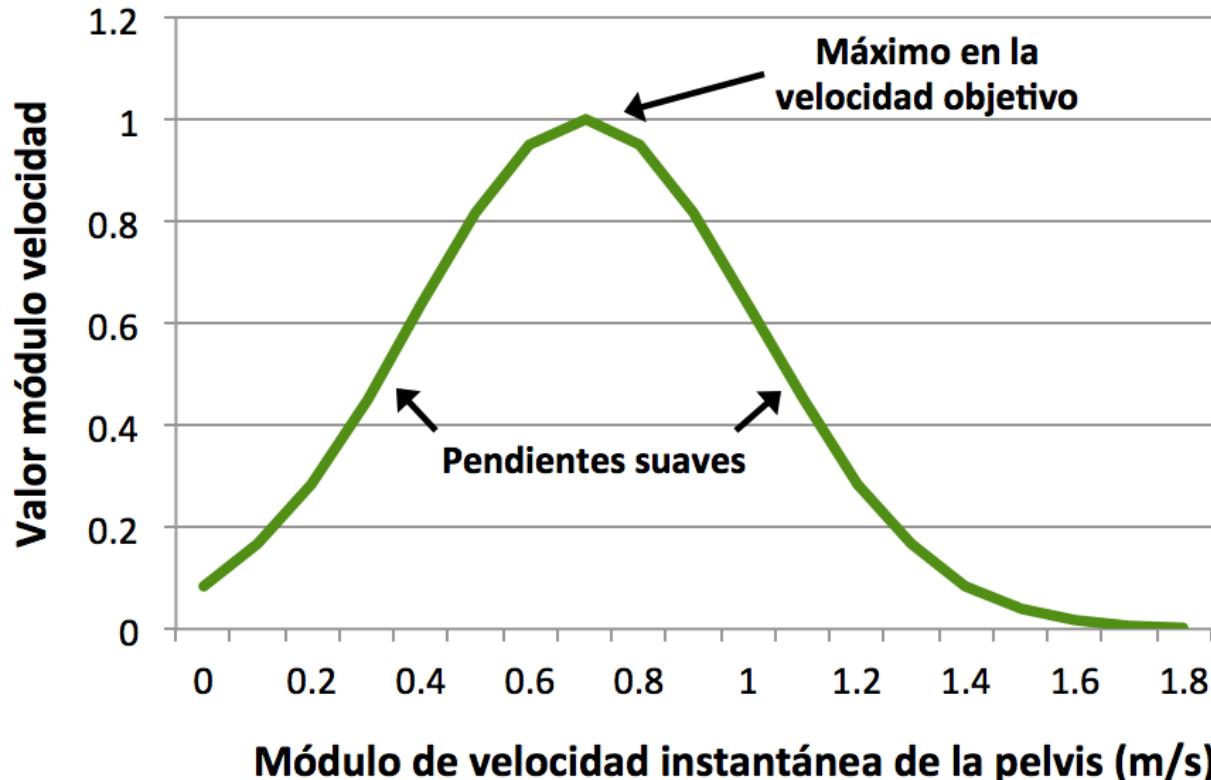
$$F = H \cdot V \cdot D \cdot S \cdot PA$$

Fitness - Altura



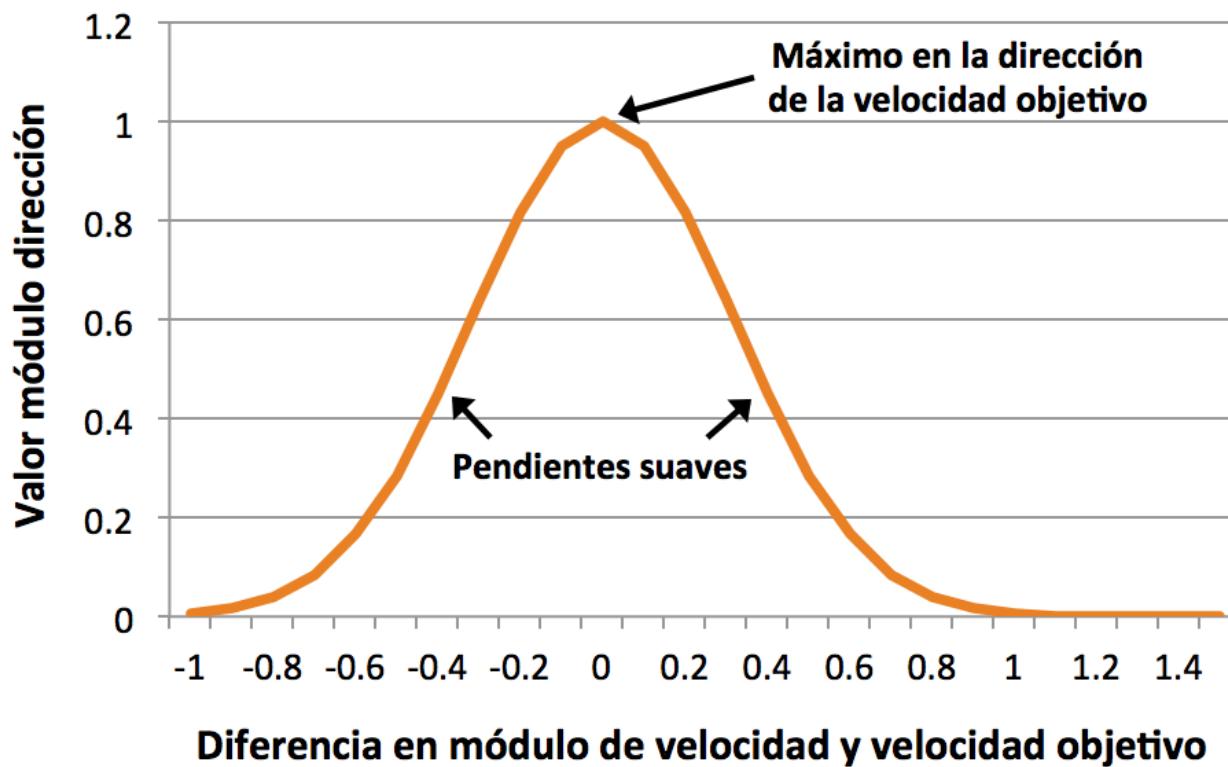
- Penaliza la caída
- Considera altura inicial
- Promedio de la simulación

Fitness - Velocidad



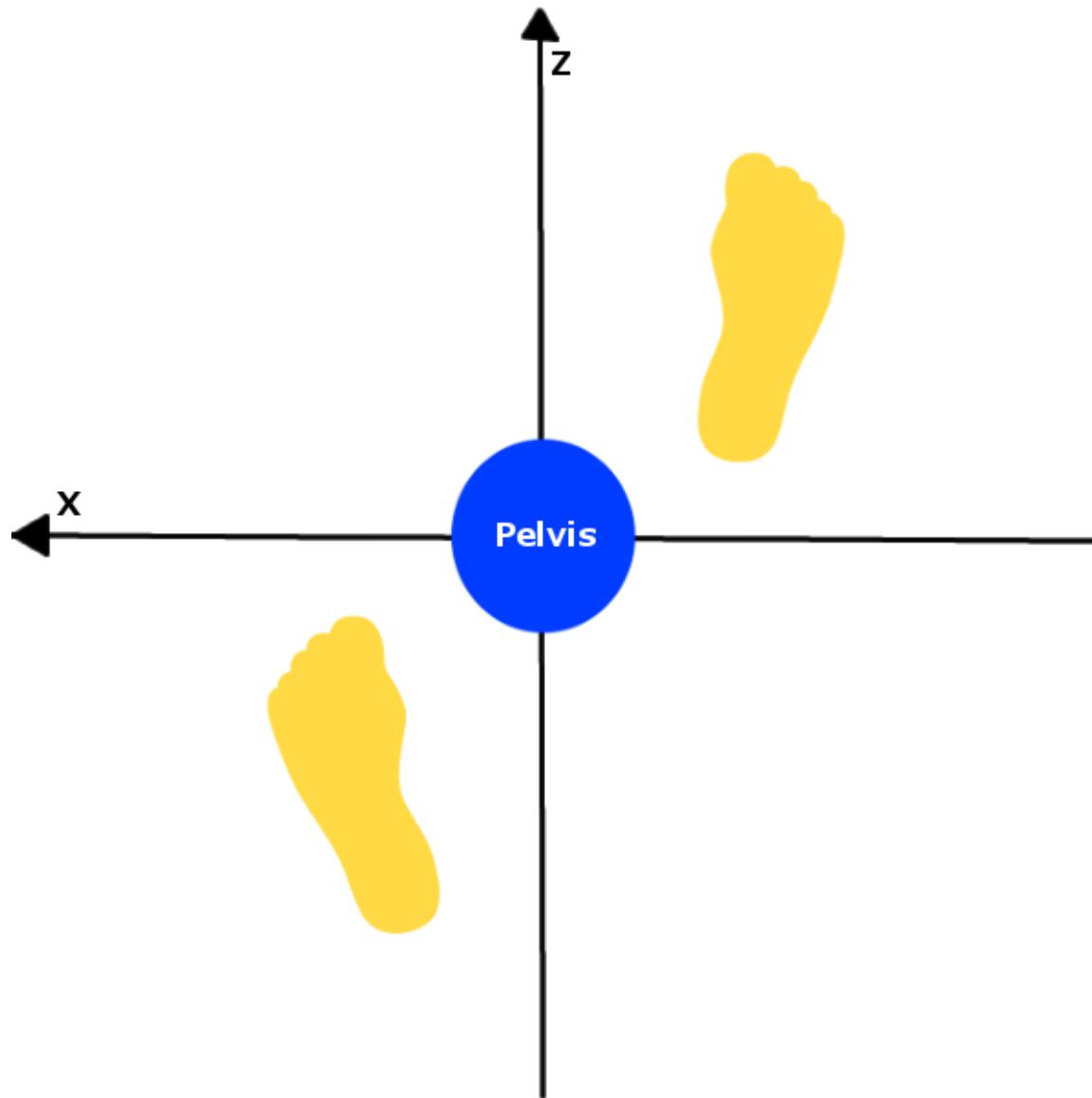
- Velocidad objetivo: 0.7 m/s
- Considera velocidad en módulo
- Promedio de la simulación

Fitness - Dirección



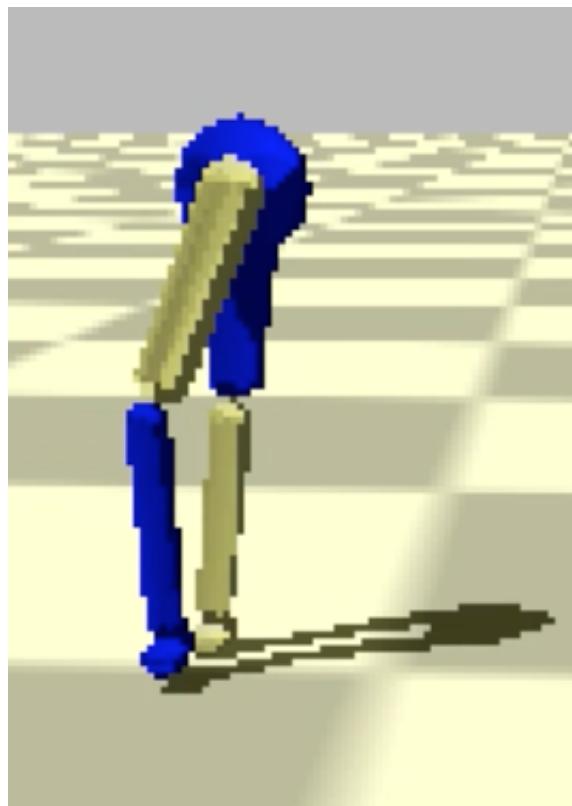
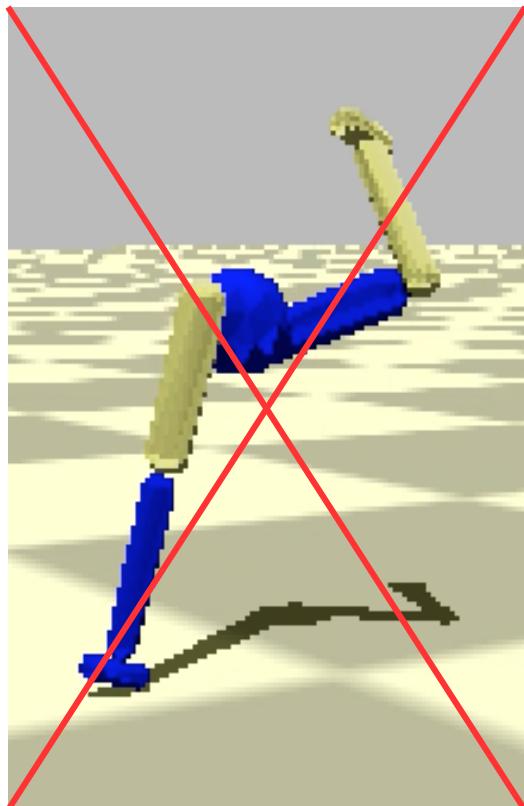
- Dirección de la velocidad
- Penaliza si es distinta del eje Z
- Complementa al módulo anterior
- Promedio de la simulación

Fitness - Simetría



- Prioriza equidistancia
- Eje de simetría: pelvis
- Considera sólo los pies
- Esencial en el ciclo de la caminata

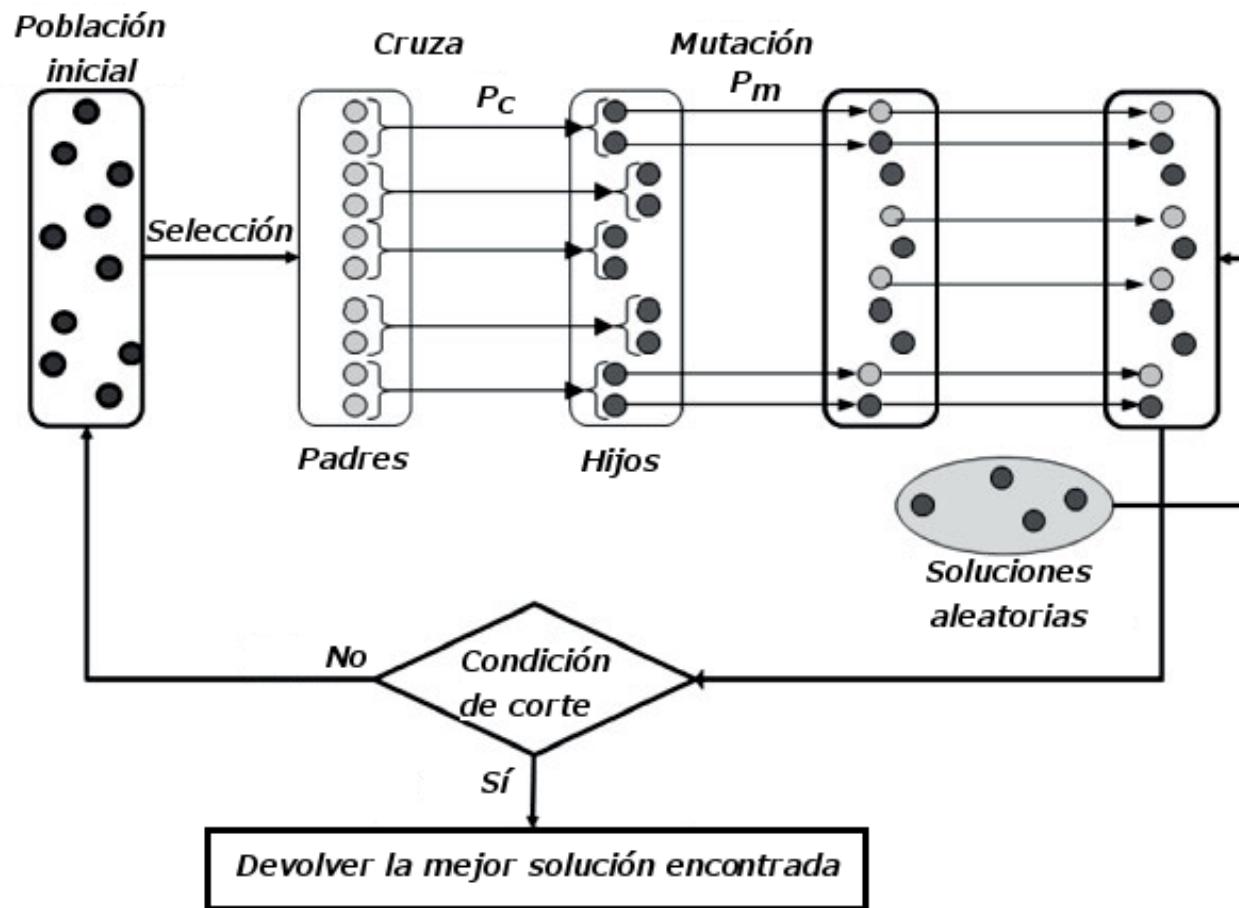
Fitness - *Pies abajo*



- Propiedad negativa
- Castiga si los pies superan la pelvis
- Pies lo más abajo posible

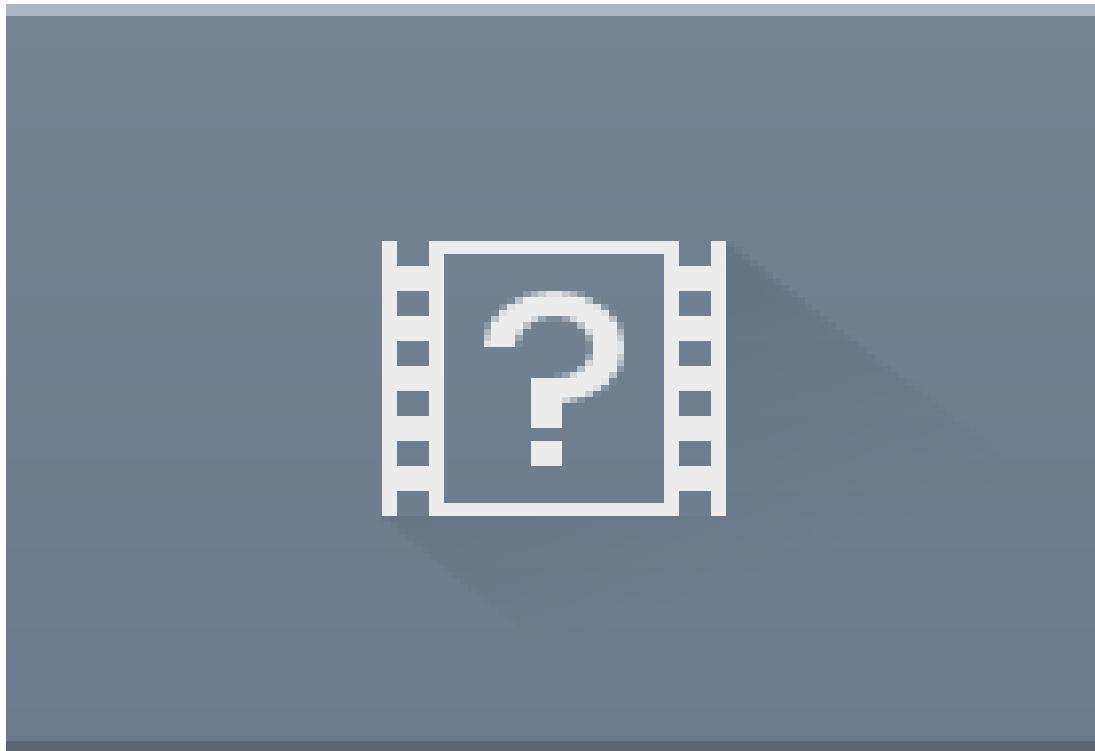
Algoritmo genético - Operadores

Configuración del algoritmo genético



- Selección: Roulette
- Elitismo activado
- $P_c: 0.9$
- $P_m: 0.3$
- 1000 generaciones
- 50 individuos por generación

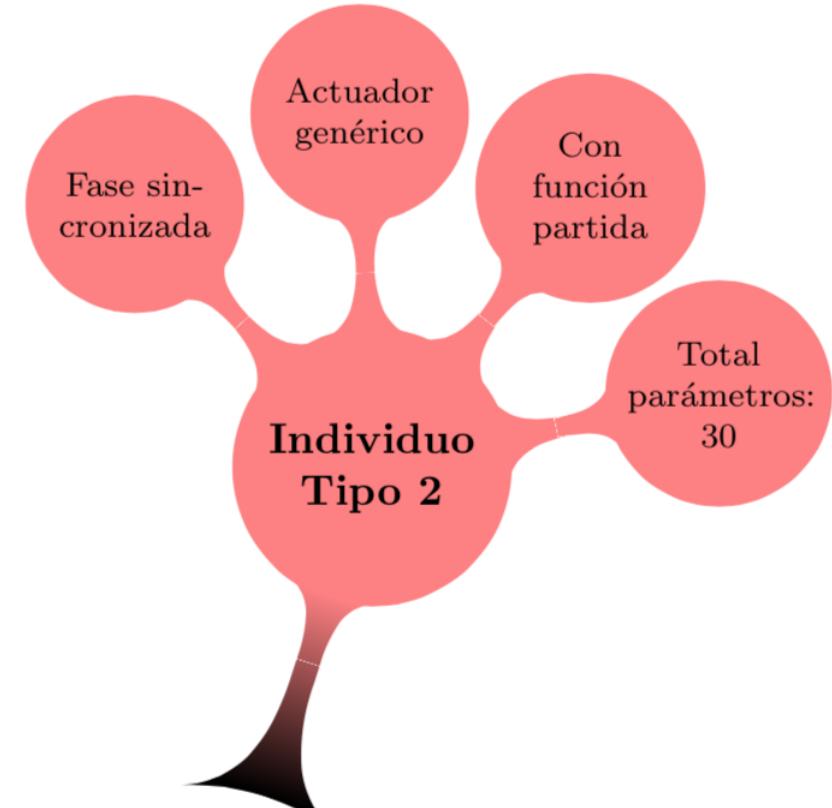
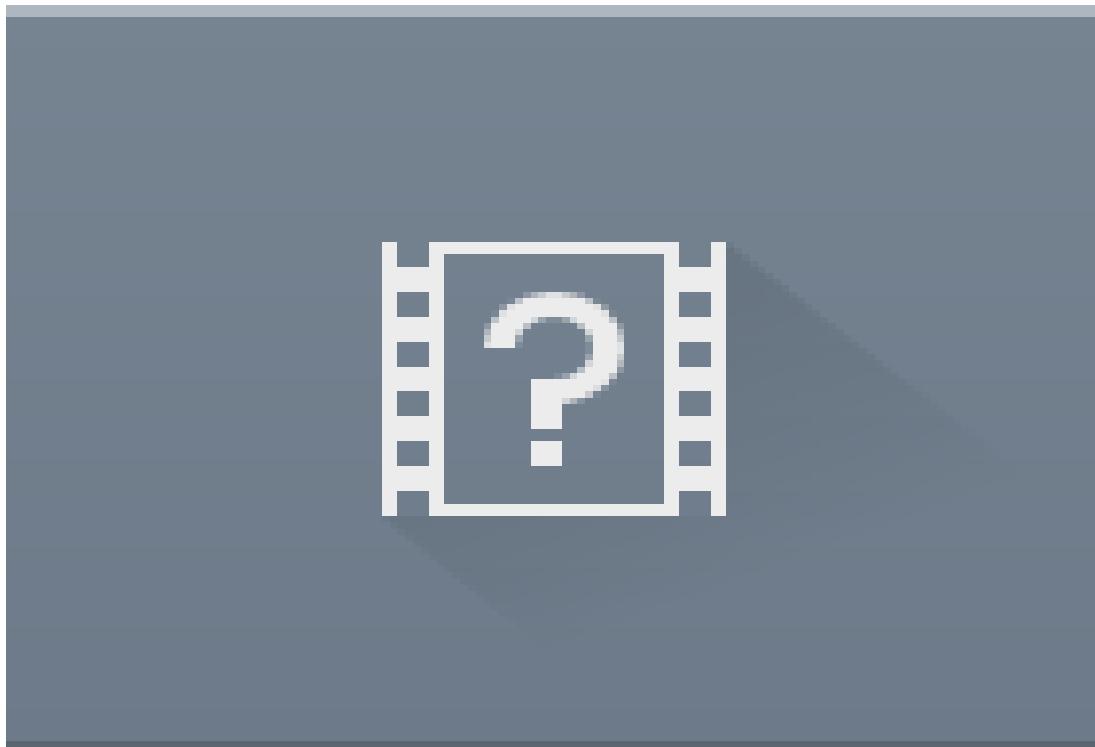
Resultados obtenidos



Comportamiento

- Fitness alto
- Da varios pasos
- Caminata poco natural

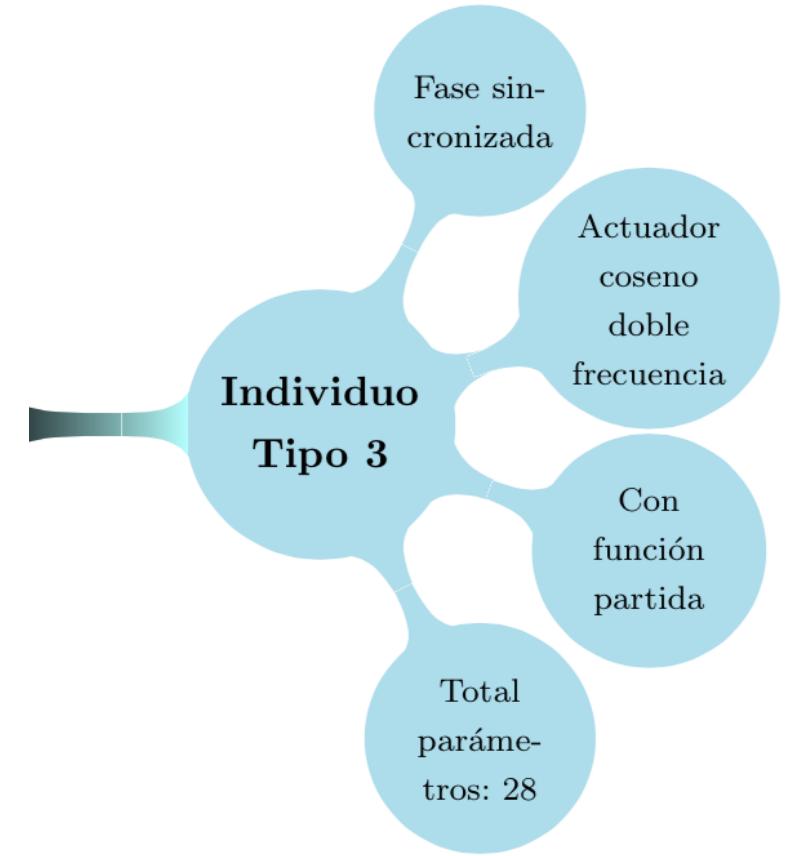
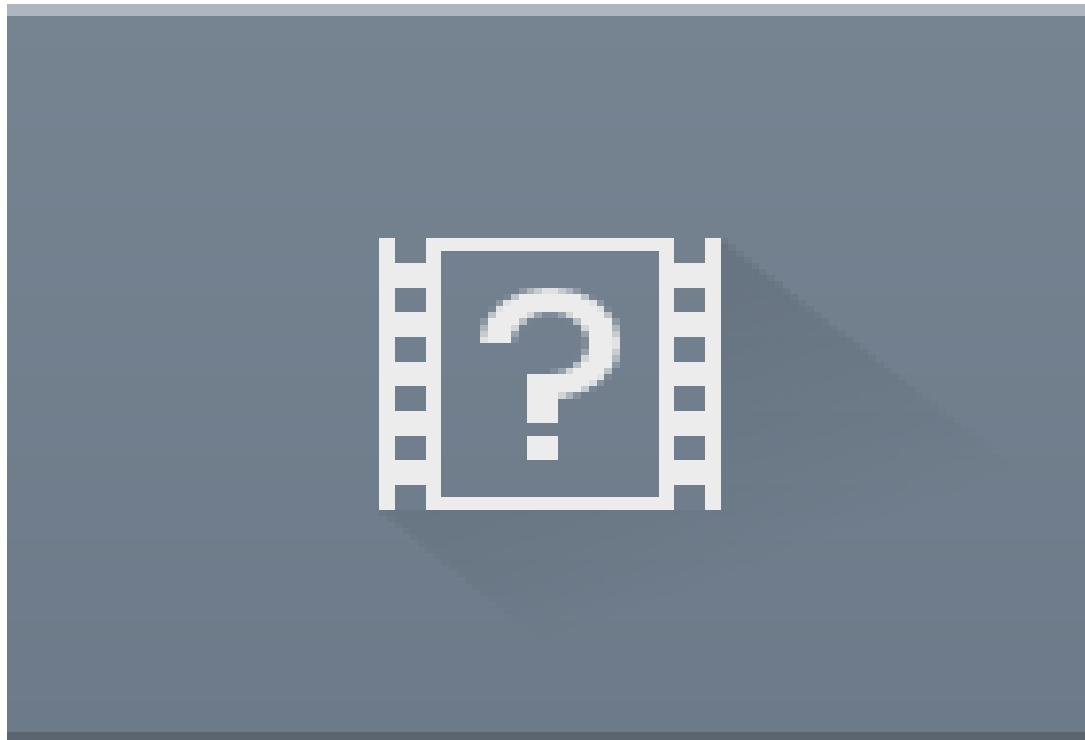
Resultados obtenidos



Comportamiento

- Fitness alto
- Caminata natural

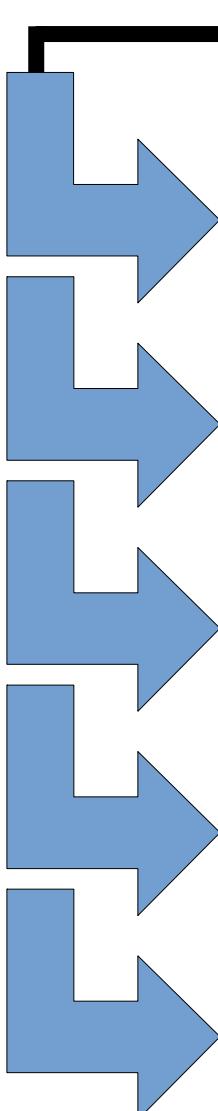
Resultados obtenidos



Comportamiento

- Fitness alto
- Muy estable
- “Passive walker”

Conclusiones



- Caminata humanoide lograda.
- El modelo del cuadrúpedo sirvió.
 -
- Restricciones de los grados de libertad.
 -
- Función partida mejora caminata.
- Actuadores decisivos.

¡Gracias!