

1

2

PROYECTO FINAL

3

INGENIERÍA INFORMÁTICA - ITBA

4

5

SIMULACIÓN Y ANIMACIÓN BIOMECÁNICA DE UN HUMANOIDE

6

7

Alumnos:

8

Enzo Altamiranda Graterol

9

ealtamir@itba.edu.ar

10

Teresa Fontanella De Santis

11

tfontane@itba.edu.ar

12

Tomás Mehdi

13

tmehd@itba.edu.ar

14

Tutor:

15

Dr. Daniel Ricardo Parisi

16

Instituto Tecnológico de Buenos Aires - ITBA

17

Departamento de Ingeniería Informática

18

Marzo 2016

Índice

19

20	Resumen	4
21	1 Introducción	4
22	2 Herramientas	5
23	2.1 Motor Físico	5
24	2.1.1 Funcionamiento	5
25	2.1.2 Modelo de fricción y su verificación	5
26	2.1.2.1 Verificación del coeficiente de fricción	6
27	2.1.2.2 Verificación del coeficiente de restitución	10
28	2.1.3 Ventajas	11
29	2.1.4 Desventajas	11
30	2.2 Librería de Algoritmos Genéticos	12
31	2.3 Código Fuente	12
32	3 Modelo Utilizado	12
33	3.1 Composición Física del Humanoide	13
34	3.2 Articulaciones	13
35	4 Actuadores	14
36	4.1 Genérico	14
37	4.2 Fourier de orden 2	14
38	4.3 Fourier de orden 9	15
39	4.4 Doble seno	15
40	5 Condiciones iniciales y de contorno	15
41	5.1 Función Partida	15
42	5.2 Fase sincronizada	16
43	6 Algoritmo Genético	16
44	6.1 Individuo	16
45	6.1.1 Parámetros	16
46	6.1.2 Valores	16
47	6.1.3 Implementación de individuos	17
48	6.1.4 Constitución del cromosoma	17
49	6.2 Fitness	18
50	6.2.1 Altura	18
51	6.2.2 Velocidad	19
52	6.2.3 Dirección	19
53	6.2.4 Simetría	19
54	6.2.5 Pies abajo	20
55	6.3 Parámetros del Algoritmo	20
56	6.3.1 Métodos de selección	20
57	6.3.2 Métodos de cruce	20
58	6.3.3 Mutación	20

59	7 Resultados Obtenidos	21
60	7.1 Evolución del <i>fitness</i> según tipo de individuo	21
61	7.2 Velocidad según tipo de individuo	21
62	7.3 Altura según tipo de individuo	22
63	7.4 Comparación de tipos de individuo	23
64	7.5 Video	23
65	8 Conclusiones	23
66	Referencias	23

67

Resumen

68

69 Este proyecto tiene como objetivo crear una simulación y animación de un humano
70 virtual, con las siguientes propiedades:

71

- 72 – Biomecánica: que tanto su estructura (peso, altura y posición de cada una de sus
73 partes) como su interacción con el entorno, respondan a comportamientos físicos
reales y exactos.
- 74 – Inteligencia Artificial: que aprenda a caminar por sí mismo, utilizando para ello
75 métodos de *soft computing* como Algoritmos Genéticos.

76

1. Introducción

77

78 Siempre ha sido de interés la simulación biomecánica de seres vivos, especialmente en las
ciencias naturales (zoológia, medicina, etc.).

79

80 Ahora bien, últimamente se ha incrementado el interés en otras áreas de aplicación, como los
videojuegos (creación de personajes con reacciones más reales), y la ingeniería (verbigracia: diseño
81 de espacios cerrados, con mayores medidas de seguridad).

82

83 Una característica muy importante de este trabajo es que el humanoide no es fruto de una
animación, sino un objeto compuesto de segmentos físicos, que interactúan entre sí y con el
84 entorno, por medio de las leyes físicas; agregando así realismo a la situación simulada. La otra
85 propiedad es que el bípedo aprenda por sí solo a caminar en línea recta, sin tener en su trayectoria
86 ningún obstáculo.

87

88 Este problema se puede abordar de diversas maneras, involucrando: redes neuronales¹, sistemas
de control² (*passive walkers*³), algoritmos genéticos, entre otras. Pero varios de ellos implican
89 modelos teóricos complejos de humanoide (considerando músculos con distintos materiales, etc.).

90

91 En este proyecto se buscó aplicar algoritmos genéticos, y lograr la caminata usando un modelo de
humanoide basado en cuerpos rígidos, unidos por articulaciones, y cuyo desplazamiento depende
92 de torques aplicados a dichos cuerpos.

93

94 El presente informe, describe y analiza pormenorizadamente: en la sección 2, las herramientas
aplicadas; en la 3, el modelo del humanoide utilizado; en la 4 y 5, los diferentes tipos de actuadores
95 y funciones de partida y contorno; en la 6, el algoritmo genético; y en la 7 y 8, los resultados
96 obtenidos con sus respectivas conclusiones.

¹Una red neuronal es un paradigma de aprendizaje automático. Se trata de un sistema de interconexión de neuronas que colaboran entre sí, para producir un estímulo de salida. Dada una entrada del sistema, se produce una salida, originada por varias transformaciones intermedias.

²definir sistema de control

³Un *passive walker* utiliza el movimiento natural (*swinging*) de las piernas para ahorrar energía usada por motores. Para caminar, calcula la posición de ciertos puntos (las articulaciones, mayormente).

⁹⁷ 2. Herramientas

⁹⁸ 2.1. Motor Físico

⁹⁹ Se le llama motor físico o *physics engine* a un “*software* capaz de realizar simulaciones de
¹⁰⁰ ciertos sistemas físicos, como la dinámica del cuerpo rígido, el movimiento de un fluido y la
¹⁰¹ elasticidad”[1].

¹⁰² Actualmente, existen muchos motores físicos: ya sea de código propietario (*PhysX*, *Havok*), como
¹⁰³ *open-source* (*Bullet Physics*, *Box2D*, *Newton*, *OGRE*). Considerando análisis relacionados [2][3],
¹⁰⁴ y el hecho de que el espacio simulado fuese en 3D, se decidió que *Bullet Physics*[4] es el más
¹⁰⁵ idóneo. Está implementado en C++ y ha sido utilizado en varios juegos (*Grand Theft Auto IV*
¹⁰⁶ y *V*, etc); en los efectos especiales de películas (*Hancock*, *Bolt*, etc.); y proyectos científicos, como
¹⁰⁷ la herramienta *open-source Tensegrity Robotics Toolkit* de la NASA⁴; entre otros.

¹⁰⁸ Si bien (como se verá más adelante) *Bullet* tiene problemas asociados con el coeficiente de restitu-
¹⁰⁹ ción, posee una muy buena *performance* en la detección de colisiones, la dinámica y la resolución
¹¹⁰ de *constraints*. Esto se debe, en parte, a diferentes algoritmos iterativos de orden lineal (donde
¹¹¹ el más importante es *Sequential Impulse*), de *caching* y también a la utilización de un modelo
¹¹² de fricción de Coulomb aproximado [5]. Además, el motor físico brinda la posibilidad de regular
¹¹³ la precisión requerida en estos cálculos (sin olvidar que, con iguales recursos, a mayor precisión,
¹¹⁴ mayor capacidad de cómputo requerida y, ergo, mayor tiempo). Dado que la construcción del hu-
¹¹⁵ manoide implica definir características y restricciones de movimiento de cada una de sus partes,
¹¹⁶ lo antes mencionado fue crucial para la elección de *Bullet Physics* en este proyecto.

¹¹⁷ 2.1.1. Funcionamiento

¹¹⁸ El motor físico se encarga de la simulación de cuerpos rígidos y la interacción entre los mismos.
¹¹⁹ En particular debe: calcular el resultado de colisiones, arreglar el solapamiento de los cuerpos
¹²⁰ en el espacio de simulación, estimar las fuerzas producidas debido a la fricción, y mantener el
¹²¹ cumplimiento de restricciones que puedan existir entre los cuerpos (por ejemplo, un vínculo para
¹²² formar una articulación). Para lograrlo, *Bullet* modela, a partir de un conjunto de ecuaciones,
¹²³ las distintas restricciones que deben ser respetadas.

¹²⁴ Estos modelos reciben como dato la velocidad lineal y angular de cada objeto, y las fuerzas
¹²⁵ que actúan sobre los mismos. Dada esta información, se resuelve el sistema de ecuaciones, cuya
¹²⁶ solución representa las magnitudes de las fuerzas a accionar sobre el mismo, a fin de satisfacer
¹²⁷ todas las restricciones. Para encontrar esta solución, entran en juego los distintos métodos de
¹²⁸ complejidad lineal mencionados en la subsección anterior.

¹²⁹ Este procedimiento, se lleva a cabo en cada *timestep* de la simulación; donde un *timestep* es el
¹³⁰ intervalo de tiempo que transcurre entre un cálculo de magnitudes y otro. Mientras menor sea
¹³¹ el *timestep*, el simulador será más preciso pero también consumirá más tiempo de cómputo.

¹³² 2.1.2. Modelo de fricción y su verificación

¹³³ Hay reglas físicas relacionadas con el entorno y que son muy importantes para la caminata:
¹³⁴ el modelo de fricción, con sus respectivos coeficientes de fricción y restitución.

⁴<http://bulletphysics.org/Bullet/phpBB3/viewtopic.php?f=17&t=9978>

137 En base a los modelos físico-matemáticos que representan a cada uno de los dos fenómenos
 138 en cuestión (y que se explicarán a continuación), y pensando en posibles futuras simulaciones
 139 de varios humanoides chocando e interactuando entre sí; se llevaron a cabo dos ensayos para
 140 verificar el funcionamiento del simulador físico *Bullet*.

141 **2.1.2.1. Verificación del coeficiente de fricción**

142 Para determinar el modelo utilizado por *Bullet* para simular las fuerzas resultantes sobre un
 143 cuerpo por acción de la fricción, se simuló un cubo, de $m_{cube} = 1\text{kg}$ y $l_{cube} = 1\text{m}$, que tiene una
 144 velocidad inicial constante (v_i) en el eje horizontal, que gradualmente se detiene por acción de
 145 la fricción, hasta llegar al reposo (Fig. 1).

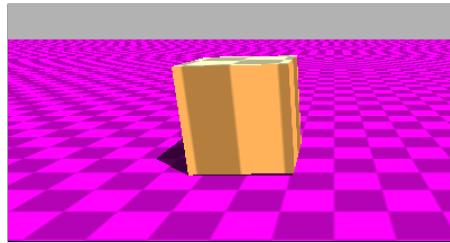


Figura 1: Visualización del sistema del cubo

146 Para este experimento se utilizó el modelo matemático que representa la posición del cuerpo
 147 en el eje horizontal en función del tiempo t , representado por la siguiente ecuación:

$$x(t) = x_i + v_i t + \frac{1}{2} a t^2 \quad (1)$$

148
 149 En este caso, el cuerpo empieza su movimiento en el origen, por lo tanto la posición inicial (x_i)
 150 es cero. v_i es la velocidad inicial, y a , la aceleración. Debido a la fricción entre el cuerpo y el
 151 suelo, se genera una fuerza de rozamiento \mathbf{F}_{μ_d} ⁵ (ec. (2)) en la misma dirección que la velocidad
 152 del sólido y en sentido contrario.

153

$$-\mathbf{F}_{\mu_d} = \mu_d \mathbf{F}_N \quad (2)$$

154 donde $\mathbf{F}_N = mg$ es la fuerza normal que actúa sobre la caja de masa m por acción de la gravedad
 155 g , y μ_d es el coeficiente de fricción dinámico.

156
 157 Finalmente, se obtiene la aceleración:

$$a = \frac{\mathbf{F}_{\mu_d}}{m} = \frac{-\mu_d \mathbf{F}_N}{m} = \frac{-\mu_d mg}{m} = -\mu_d mg \quad (3)$$

158 Considerando las ec. (1) y (3), se puede obtener el modelo matemático que predice el movimiento
 159 de la caja:

$$x(t) = x_i + v_i t - \frac{1}{2} \mu_d g t^2 \quad (4)$$

160

161

⁵En las ecuaciones, los vectores se escriben en negrita. Y las fuerzas son vectores.



Figura 2: Resultados logrados de simular el sistema descripto en Fig. 1, usando $v_i = 10 \frac{m}{s}$ y:
 (a) $\mu_d = 0.25$, (b) $\mu_d = 0.50$, y (c) $\mu_d = 0.75$



Figura 3: Resultados logrados de simular el sistema descripto en Fig. 1, usando $v_i = 3 \frac{m}{s}$ y:
 (a) $\mu_d = 0.25$, (b) $\mu_d = 0.50$, y (c) $\mu_d = 0.75$

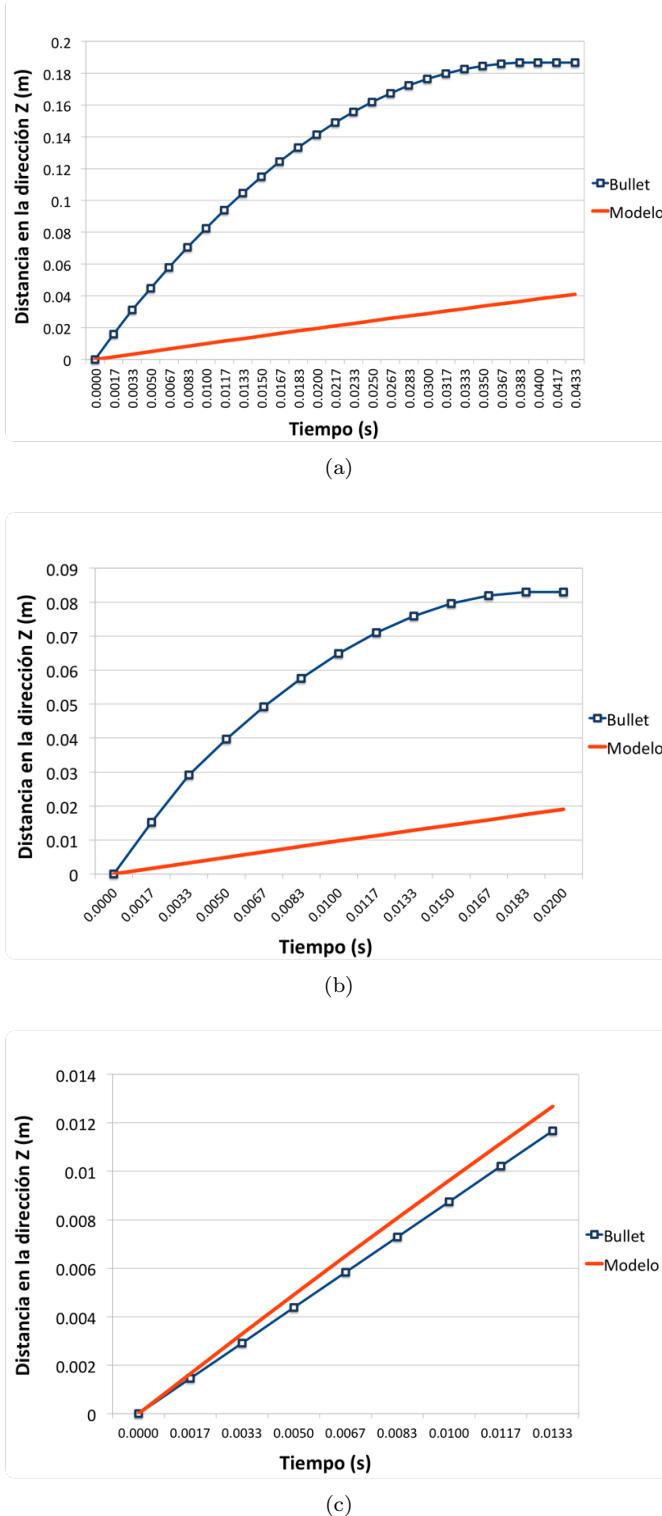


Figura 4: Resultados logrados de simular el sistema descripto en Fig. 1, usando $v_i = 1 \frac{m}{s}$ y:

(a) $\mu_d = 0.25$, (b) $\mu_d = 0.50$, y (c) $\mu_d = 0.75$

162 Los gráficos (Fig. 2, 3 y 4) demuestran que los resultados de las pruebas fueron favorables.
 163 Los valores obtenidos a partir de la simulación se corresponden con los alcanzados a partir del
 164 modelo matemático. Esto es un indicador de que *Bullet* debe estar usando dichos modelos para
 165 ejecutar las simulaciones.

166 No obstante, cabe destacar que, los gráficos que corresponden a la Fig. 4, presentan una discre-
 167 parancia mayor entre la simulación y el modelo. Este hecho puede deberse a que cuanto menor sea
 168 la cantidad de puntos tomados, menor será la resolución de la simulación, y mayor será el error
 169 cometido.

170 **2.1.2.2. Verificación del coeficiente de restitución**

171 El segundo ensayo simula una esfera a una altura determinada sobre el suelo, que tiene una
 172 velocidad v_i en el eje perpendicular al piso y que eventualmente colisiona contra el mismo. Se
 173 desea comprobar que la colisión entre el cuerpo y el suelo respete que la velocidad final v_f de
 174 la esfera después del choque sea proporcional a su coeficiente de restitución e dado por la ecuación:

175

$$e = \frac{v_f}{v_i} \quad (5)$$

176

177 Para efectuar la colisión con el suelo, se empleó una esfera sólida ubicada a 4 metros del suelo,
 178 cuya masa y radio son $m_{sphere} = 1 \text{ kg}$ y $r_{sphere} = 1 \text{ m}$, respectivamente (ver Fig. 5). A la esfera
 179 se le asigna, además, un coeficiente de restitución determinado.

180 Se eligió un ambiente sin gravedad ($g = 0 \frac{\text{m}}{\text{s}^2}$). De esta forma, se podrá tener en cuenta sólo la
 181 velocidad inicial (v_i) y la velocidad final (v_f) para el cálculo del coeficiente de restitución (e)
 182 (ver ec. (5)).

183 El intervalo de tiempo físico (o *timestep*) utilizado es $\Delta t = 0.001 \text{ s}$. El *timestep* de animación (es
 184 decir, cada cuánto tiempo se guardan en un archivo los datos logrados) es $\Delta t' = 0.1$ y el tiempo
 185 de simulación es de $s = 100\Delta t$. El coeficiente de fricción es $\mu = 0.75$.

186

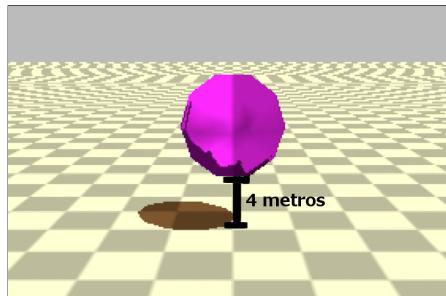


Figura 5: Visualización del sistema de la esfera

187 El ensayo tiene como parámetros de entrada: v_i (velocidad inicial) y e_{sim} (coeficiente de res-
 188 titución esperado). Por otro lado, se obtiene v_f (velocidad de la esfera al finalizar la simulación);
 189 y se calculan e_{medida} (coeficiente de restitución obtenido a partir de la ec. (5)) y ϵ_{rel} , que es el
 190 error relativo entre los coeficientes e_{sim} y e_{medida} , calculado de la siguiente manera:

191

$$\epsilon_{rel} = \frac{e_{sim} - e_{medida}}{e_{sim}} \quad (6)$$

192

193 Se muestra en la Tabla 1 lo arrojado por los experimentos numéricos efectuados. En ellos,
 194 $v_i = \{-0.5, -3.5, -4, -5 \text{ y } -10\} \frac{\text{m}}{\text{s}}$ y $e_{sim} = \{0.2, 0.5 \text{ y } 0.8\}$.

v_i	e_{sim}	v_f	e_{medida}	ϵ_{rel}
$-0.5 \frac{m}{s}$	0.2	$0.000249 \frac{m}{s}$	0.000498	0.997
	0.5	$0.000219 \frac{m}{s}$	0.000438	0.999
	0.8	$0.001037 \frac{m}{s}$	0.002074	0.997
$-3.5 \frac{m}{s}$	0.2	$0.000057 \frac{m}{s}$	0	1
	0.5	$0.000018 \frac{m}{s}$	0	1
	0.8	$0.3 \frac{m}{s}$	0.0857	0.893
$-4 \frac{m}{s}$	0.2	$0.000473 \frac{m}{s}$	0.00012	1
	0.5	$0.000424 \frac{m}{s}$	0.00011	1
	0.8	$1.23 \frac{m}{s}$	0.3	0.625
$-5 \frac{m}{s}$	0.2	$1 \frac{m}{s}$	0.2	0
	0.5	$2.5 \frac{m}{s}$	0.5	0
	0.8	$4 \frac{m}{s}$	0.8	0
$-10 \frac{m}{s}$	0.2	$2 \frac{m}{s}$	0.2	0
	0.5	$5 \frac{m}{s}$	0.5	0
	0.8	$8 \frac{m}{s}$	0.8	0

Tabla 1: Coeficientes de restitución obtenidos de simular el sistema descrito en Fig. 5

195 Los resultados exponen una limitación del motor físico: no representa correctamente las colisiones elásticas entre esferas y cuerpos rígidos, que ocurren a velocidades bajas. Esto queda en evidencia en la Tabla 1. En cada una de ellas el error fue de casi 1. La razón por la que ocurre este hecho se debe a que *Bullet* utiliza un algoritmo de colisión que frena la velocidad de un objeto que está a punto de colisionar. Haciendo esto puede evitar que los sólidos se traspasen y de esta forma se pueden realizar cálculos de fuerza más precisos.

201 En el caso de los ensayos, las esferas poseen una velocidad muy baja, cuando están a punto de colisionar *Bullet* reduce aún más esta velocidad y eventualmente quedan con una velocidad tan baja que al chocar contra el suelo se aplica el efecto restitutivo a esta velocidad casi nula y se resuelve que la esfera debe quedar en reposo, cuando en realidad debería poseer una velocidad baja, pero no despreciable.

206 2.1.3. Ventajas

207 Las ventajas del motor físico son:

- 208 – Código abierto: mayor conocimiento sobre las fórmulas y métodos implementados en el motor.
- 209 – Soporte de la comunidad científica.
- 210 – Licencia libre.

211 2.1.4. Desventajas

212 Como toda herramienta, *Bullet* tiene aspectos negativos, entre los que se encuentran:

- 213 – Documentación poco clara y desordenada.
- 214 – Debido a que la física se aproxima usando métodos numéricos que contienen error, las simulaciones son no determinísticas.

- 216 – Utilizar una librería gráfica (como *OpenGL*) acoplada a una simulación de *Bullet* puede
217 producir resultados distintos, que si se usa un programa de visualización externo (como
218 OVITO).

219 **2.2. Librería de Algoritmos Genéticos**

220 Se utilizó la conocida librería de Algoritmos Genéticos para C++ GaLib, desarrollada por
221 Matthew Wall del MIT [6].

222 Ofrece funcionalidades como: programación paralela, diversos métodos de selección (*elite*, *rou-*
223 *lette*), estrategias de reemplazo (de padres, aleatorio, del peor), entre otras.

224 **2.3. Código Fuente**

225 Al estar *Bullet* implementado en C++, el código fuente también está desarrollado en ese
226 lenguaje. En *Bullet*, se define un *World* (o mundo físico) en donde se puede insertar, entre otras
227 cosas, cuerpos rígidos. En este caso en particular, el mundo consta de un plano (el suelo) y el
228 humanoide encima (compuesto por cuerpos rígidos y otros elementos físicos).

229 El *software* creado incluye: creación del humanoide, siendo éste capaz de desplazarse por medio
230 de actuadores (que se verán en la Sección 4); el desarrollo del algoritmo genético (la definición de
231 los individuos, la función de *fitness*, métodos de selección, etc.); visualización gráfica del mejor
232 humanoide logrado por el algoritmo genético; y la posibilidad de realizar gráficos referidos a la
233 evolución del algoritmo genético (*fitness* por cada generación, etc.).

234 Se acompañan a esta presentación: el código fuente y el manual de instalación y uso.

235 **3. Modelo Utilizado**

236 Dentro de los diversos modelos existentes ([7][8]); en este proyecto se procuró utilizar uno que
237 fuera sencillo pero representativo a la vez.

238 Se modela al cuerpo humano, con el motor *Bullet Physics*, como un conjunto de segmentos unidos
239 por articulaciones (Fig. 6). A cada uno de ellos se les aplica una fuerza en el centro de masa de
240 cada segmento (denominada Actuador). Que la caminata se produzca o no, depende del tipo de
241 actuador utilizado (la función utilizada para la fuerza), y de sus parámetros. El objetivo, entonces,
242 se reduce a encontrar dichos parámetros. Para eso se usan los algoritmos genéticos, un método
243 de Inteligencia Artificial. De este modo se obtiene, de forma análoga a la selección natural, los
244 individuos que mejor se adapten a la caminata. Tanto los actuadores como el algoritmo genético
245 se explicarán más adelante.

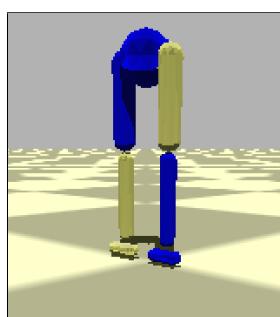


Figura 6: Humanoide diseñado

246 3.1. Composición Física del Humanoide

247 Como ya se expresó, el humanoide fue modelado en *Bullet* como un conjunto de segmentos
 248 (cuerpos rígidos), unidos por articulaciones.

249 Se dividió al cuerpo humano en: cabeza, tronco, miembro superior, pelvis y miembro inferior
 250 (muslo, pierna y pie). Pero a los fines de este proyecto, sólo se consideraron la pelvis y el miem-
 251 bro inferior (ver Fig. 7).

252 A continuación se presenta la composición de cada segmento (de acuerdo a la biomecánica [9]).

253

Parte	Cantidad	Forma	Largo (en m)	Peso (en kg)	Uniones
Pelvis	1	esférico	0.08655	9.9718	Cadera
Muslo	2	esfero-cilindro	0.4015	10.3368	Cadera y Rodilla
Pierna	2	esfero-cilindro	0.4015	3.1609	Rodilla y Tobillo
Pie	2	esfero-cilindro	-	1.0001	Tobillo

Tabla 2: Segmentos del humanoide

254 3.2. Articulaciones

255 Para unir los distintos segmentos entre sí, se utilizaron articulaciones bisagra con 1 grado de
 256 libertad: en el eje X, para que los segmentos puedan moverse en dos ejes: el Z, donde ocurre la
 257 caminata, y el Y, perpendicular al piso (ver Fig. 7).

258 Asimismo, para cada caso en particular, se definieron cotas para los ángulos que pueden existir
 259 entre los segmentos. Esto es muy importante, no sólo porque se adecúa a datos biológicos, sino
 260 porque, de otro modo la caminata no podría lograrse: si los ángulos son demasiado altos, la ca-
 261 minata se produce girando las piernas por encima de la pelvis; si por el contrario, son demasiado
 262 bajos, las piernas van a estar muy rígidas, originando pocos pasos y muy cortos.

263 Por otra parte, a la pelvis se le restringe todo tipo de rotación.

264

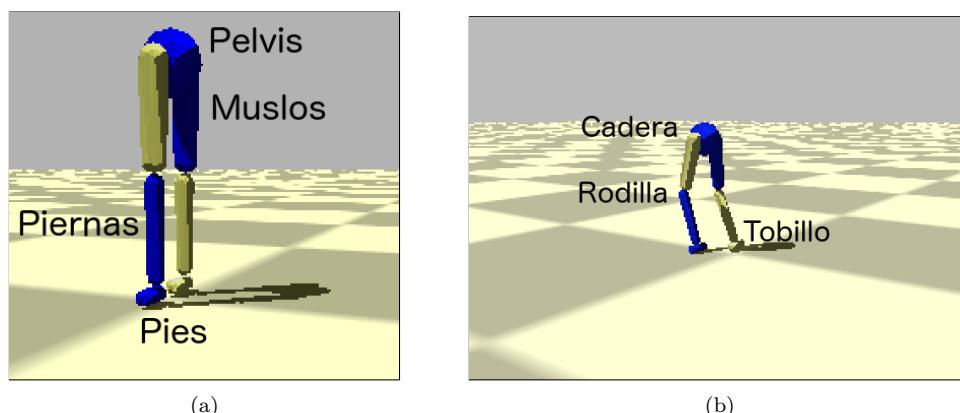


Figura 7: Humanoide diseñado: (a) segmentos, y (b) articulaciones

265 4. Actuadores

266 A cada uno de los segmentos correspondientes al muslo y la pierna del bípedo, se le aplica
 267 un torque (o actuador) en el eje X (perpendicular a la trayectoria), como se ve en Fig 8. Así,
 268 pueden moverse para arriba o para abajo (con respecto a la articulación a la que pertenecen).

269 A fin de simplificar el modelo, el humanoide tiene el mismo tipo de actuador utilizado en todos
 270 los segmentos.

271 Es necesario aclarar que la caminata producida por el humanoide es plana (en 2D y no en 3D,
 272 como debería ser en una caminata real). Esto se debe a que la trayectoria pensada para el bípedo
 273 es una línea recta, y logrando que los segmentos se muevan en un solo eje es suficiente para
 274 cumplir con dicha trayectoria. También contribuye el hecho de que el torque se aplique en una
 275 sola dimensión. Por otra parte, los actuadores definidos a continuación son periódicos, y por eso
 276 no se pueden aplicar en el eje Z de los segmentos (se necesitarían actuadores reactivos, para
 277 poder detectar cuando el humanoide se está cayendo, etc.).

278 Para indicar el módulo de dicho torque, se diseñaron diferentes funciones (todas ellas periódicas),
 279 mencionadas en las subsecciones que siguen.

280

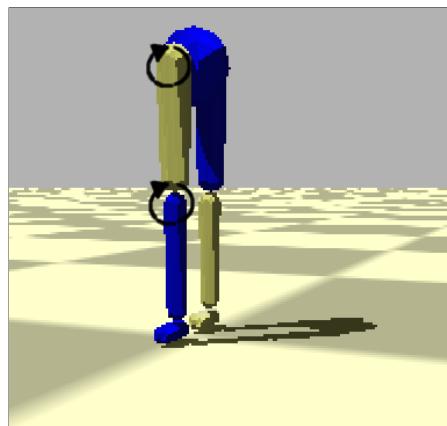


Figura 8: Aplicación de los actuadores en los segmentos del bípedo

281 4.1. Genérico

282 Es el actuador más sencillo, tanto matemática como computacionalmente.

$$f(t) = A_1 \sin(\omega_1 t + \phi) + A_2 \cos(\omega_2 t + \phi) + C \quad (7)$$

283 donde $f(t)$ es la función del actuador evaluada en el instante de tiempo t , A_1 y A_2 son amplitudes,
 284 ω_1 y ω_2 son frecuencias (en $\frac{1}{s}$), ϕ es la fase (en radianes), y C es un término independiente.

285 La fase ϕ es la misma en el seno y en el coseno, para evitar que se formen otro tipo de funciones
 286 no cíclicas.

287 4.2. Fourier de orden 2

288 Este actuador utiliza una serie de Fourier de dos términos.

$$f(t) = A_1 \sin(\omega t + \phi) + B_1 \cos(\omega t + \phi) + A_2 \sin(2\omega t + \phi) + B_2 \cos(2\omega t + \phi) + C \quad (8)$$

289 donde A_1 , A_2 , B_1 y B_2 son amplitudes y ω es frecuencia (en $\frac{1}{s}$).

290 4.3. Fourier de orden 9

291 Es una extensión del actuador anterior, pero con 9 términos. Por ser de mayor grado, brinda
 292 una mayor precisión. Sin embargo, es más difícil de manejar computacionalmente; y, además,
 293 que sea más preciso no garantiza que con él se pueda lograr una buena caminata.

$$294 \quad f(t) = A_1 \sin(\omega t + \phi) + B_1 \cos(\omega t + \phi) + A_2 \sin(2\omega t + \phi) + B_2 \cos(2\omega t + \phi) \\ + A_3 \sin(3\omega t + \phi) + B_3 \cos(3\omega t + \phi) + A_4 \sin(4\omega t + \phi) + B_4 \cos(4\omega t + \phi) \\ + A_5 \sin(5\omega t + \phi) + B_5 \cos(5\omega t + \phi) + A_6 \sin(6\omega t + \phi) + B_6 \cos(6\omega t + \phi) \quad (9) \\ + A_7 \sin(7\omega t + \phi) + B_7 \cos(7\omega t + \phi) + A_8 \sin(8\omega t + \phi) + B_8 \cos(8\omega t + \phi) \\ + A_9 \sin(9\omega t + \phi) + B_9 \cos(9\omega t + \phi) + C$$

295 donde A_i y B_i con $1 \leq i \leq 9$ son amplitudes.

296 4.4. Doble coseno

297 Esta función periódica utiliza medio ciclo de una función sinusoidal, y medio ciclo de otra
 298 (ambas pueden tener frecuencias distintas). De esta manera, se consigue una caminata más
 299 natural, y que no ocurre con los actuadores de Fourier, que producen una doble flexión de las
 300 rodillas en cada ciclo.

301 La idea es lograr una función periódica a partir de una que no lo es (ya que t es lineal). Para
 302 eso, se utiliza $\psi(t)$ (ec. (10)) que aplica una transformación a los números reales, para que se
 303 encuentren dentro del rango del ciclo completo (con las dos frecuencias). ω es la frecuencia de
 304 $f(t)$ (ec. (12)), que utiliza medio ciclo con frecuencia ω_1 y medio ciclo con frecuencia ω_2 .

$$\psi(t) = t + \phi - \left\lfloor \frac{t + \phi}{\pi/\omega_1 + \pi/\omega_2} \right\rfloor (\pi/\omega_1 + \pi/\omega_2) \quad \psi : \mathbb{R} \rightarrow \left[0, \frac{2\pi}{\omega} \right] \quad (10)$$

305

306

$$\omega = \frac{2\omega_1\omega_2}{\omega_1 + \omega_2} \quad (11)$$

307

308

$$f(t) = \begin{cases} A \cos(\omega_1 \psi(t)) + C & \text{si } \omega_1 \psi(t) < \pi \\ A \cos(\omega_2(\psi(t) - (\pi/\omega_1) + (\pi/\omega_2))) + C & \text{en otro caso} \end{cases} \quad (12)$$

309 5. Condiciones iniciales y de contorno

310 Como las funciones periódicas señaladas en los actuadores, no son suficientes para lograr la
 311 caminata, se le adosaron las funciones que se detallan seguidamente.

312 5.1. Función Partida

313 El andar del humanoide es cíclico. Sin embargo, por la posición inicial del individuo, se requiere
 314 para el tiempo del primer paso, una función distinta a la del resto de la caminata. El tipo de
 315 función puede ser cualquiera de los actuadores vistos anteriormente (pero no necesariamente con
 316 los mismos valores de amplitud, frecuencia y fase asignados a las piernas). Empero, se utilizó la
 317 función vista en el actuador genérico.

³¹⁸ Por otra parte, para simplificar el modelo, se decidió que el tiempo considerado para el primer
³¹⁹ paso sea fijo, y de 0.7 segundos. Dicho valor fue extraído de forma experimental.

³²⁰ 5.2. Fase sincronizada

³²¹ En una caminata, las piernas deben guardar simetría: mientras una va hacia adelante, la otra
³²² va hacia atrás (y viceversa). Esto, de acuerdo con los actuadores definidos en la sección anterior,
³²³ implica que las funciones de movimiento de cada pierna estén desfasadas en medio ciclo ($\frac{\pi}{2}$):

$$f_i(t) = f(t) \quad (13)$$

$$f_d(t) = f(t + \frac{\pi}{2}) \quad (14)$$

³²⁵ siendo $f(t)$ la función de movimiento (o actuador) en el momento t , y f_i y f_d las funciones de la
³²⁶ pierna izquierda y derecha, respectivamente.

³²⁷ 6. Algoritmo Genético

³²⁸ 6.1. Individuo

³²⁹ La información genética de cada individuo, está compuesta por dos partes: función partida
³³⁰ (optativa) y los parámetros asociados a los actuadores (obligatorios).

³³¹ 6.1.1. Parámetros

³³² Tanto la función partida como los actuadores tienen como parámetros: amplitud (A o B);
³³³ fase (ϕ), que indica dónde comienza el paso y se mide en radianes; frecuencia (ω); y término
³³⁴ independiente (C).

³³⁵ 6.1.2. Valores

³³⁶ Cada uno de los segmentos tiene propiedades físicas distintas (masa, largo, etc.), razón por
³³⁷ la cual no necesariamente sus genes deban tener los mismos rangos de valores, tal como puede
³³⁸ apreciarse en las Tablas 3 y 4.

³³⁹

Segmento	Tipo de gen	Mínimo	Máximo
Muslo	Amplitud	-30	30
Pierna	Amplitud	-60	60
Muslo y Pierna	Frecuencia	0.01	10
Muslo y Pierna	Fase	$-\pi$	π
Muslo y Pierna	Término independiente	-10	10

Tabla 3: Rango de valores que puede tomar cada gen, para la función partida

Actuador	Segmento	Tipo de gen	Mínimo	Máximo
Genérico	Muslo	Amplitud	-30	30
	Pierna	Amplitud	-60	60
	Muslo y Pierna	Frecuencia	0.01	10
	Muslo y Pierna	Fase	$-\pi$	π
	Muslo y Pierna	Término independiente	-10	10
Fourier de orden 2	Muslo	Amplitud	-60	60
	Pierna	Amplitud	-30	30
	Muslo y Pierna	Frecuencia	0.01	10
	Muslo y Pierna	Fase	$-\pi$	π
	Muslo y Pierna	Término independiente	-10	10
Fourier de orden 9	Muslo y Pierna	Amplitud	-60	60
	Muslo y Pierna	Frecuencia	0.1	2
	Muslo y Pierna	Fase	$-\pi$	π
	Muslo y Pierna	Término independiente	-10	10
Doble Coseno	Muslo	Amplitud	-50	50
	Pierna	Amplitud	-30	30
	Muslo y Pierna	Frecuencia	0.01	5
	Muslo y Pierna	Fase	$-\pi$	π
	Muslo y Pierna	Término independiente	-30	30

Tabla 4: Rango de valores que puede tomar cada gen, según el tipo de actuador

340 6.1.3. Implementación de individuos

341 Para favorecer el análisis de las distintas características arriba indicadas, se implementaron
 342 varios individuos, cada uno de ellos con propiedades distintas (Tabla 5).

Individuos	Actuador	Función Partida	Fase Sincronizada
Tipo 1	Genérico	No	Sí
Tipo 2	Genérico	Sí	Sí
Tipo 3	Fourier de orden 2	Sí	Sí
Tipo 4	Fourier de orden 9	Sí	Sí
Tipo 5	Doble coseno	Sí	Sí

Tabla 5: Tipo de individuos

343 6.1.4. Constitución del cromosoma

344 La longitud del cromosoma de un individuo depende de los actuadores y la función partida
 345 usados. Cada uno de ellos, a su vez, tienen los parámetros presentados en la sección 6.1.1. Sus
 346 respectivas cantidades pueden verse en la Tabla 6.

347 En la Tabla 7 se muestra la composición del cromosoma de cada individuo. En ella se puede
 348 observar cómo según el tipo de individuo, varía la cantidad de genes, es decir, el tamaño del
 349 cromosoma. Vale aclarar que la función partida se especifica para cada segmento (las dos piernas
 350 y los dos muslos); en cambio, para los actuadores, sólo se definen dos (uno para las piernas y
 351 otro para los muslos).

Actuadores	Parámetros			
	Amplitud	Frecuencia	Fase	Término independiente
Genérico	2	2	1	1
Fourier de orden 2	4	1	1	1
Fourier de orden 9	18	1	1	1
Doble coseno	1	2	1	1
Función partida	2	2	1	1

Tabla 6: Cantidad de parámetros según tipo de actuador o función partida

Parámetros	Individuos				
	Tipo 1	Tipo 2	Tipo 3	Tipo 4	Tipo 5
Amplitud	12	12	16	44	8
Frecuencia	2	6	6	6	8
Fase	2	6	6	6	6
Término independiente	2	6	6	6	6
Totales	18	30	34	62	28

Tabla 7: Cantidad de parámetros según tipo de individuos

352 6.2. Fitness

353 El papel de la función de *fitness* F en un algoritmo genético es evaluar qué tan bueno es un
 354 individuo. En este caso, está definida como un producto de cinco módulos o propiedades: altura
 355 (H), velocidad (V), dirección (D), simetría (S) y pies abajo (PA):

$$F = H * V * D * S * PA \quad (15)$$

356 Los cinco tienen la misma importancia y, por eso, como se verá a continuación, están definidos de
 357 forma similar (con una función exponencial y pueden valer entre 0 y 1). Con todo esto, dado que
 358 el *fitness* está pensado como un producto, basta con que uno de los módulos sea muy chico para
 359 “anular” al individuo (es decir, otorgarle un valor que tiende a cero). Sin embargo, los diferentes
 360 módulos no son completamente independientes entre sí: por ejemplo, si la altura es demasiado
 361 baja, posiblemente la velocidad y la dirección no sean adecuadas.

362

363 6.2.1. Altura

364 Es un factor relacionado con la altura del individuo en toda la simulación, y se expresa:

$$365 H = \frac{\sum_{n=0}^T e^{-C(h_{t_n} - h_{t_0})^2}}{N} \quad (16)$$

366

367 donde t_0 es el tiempo inicial, t_T el tiempo final, h_{t_n} es la altura de la pelvis en el instante de
 368 tiempo t_n , N la cantidad pasos de simulación y C una constante $C = 5$.

369 Se calcula a partir de la diferencia entre la altura en cada instante de la simulación, con su altura
 370 inicial (la altura está definida como la posición de la pelvis en el eje Z). Cuanto mayor sea esa
 371 diferencia, más rápido el individuo cae, y por eso este módulo tiende a cero. Por el contrario,

372 valdrá uno si la diferencia es ínfima (lo que significa que el humanoide mantiene su misma altura
373 durante la caminata).

374

375 6.2.2. Velocidad

376 Indica qué tan cercana es la velocidad del individuo con respecto a una velocidad objetivo
377 (en este caso, es de 1.2 m/h), y se expresa de la siguiente forma:

378

$$V = \frac{\sum_{n=0}^T e^{-C(v_{t_n} - V_O)^2}}{N} \quad (17)$$

379

380 donde t_0 es el tiempo inicial, t_T el tiempo final, v_{t_n} es la velocidad de la pelvis en el instante de
381 tiempo t_n , y V_O la velocidad objetivo en el eje Z (el eje de la caminata).

382 Sigue una lógica y cálculo similares al factor de altura: a mayor discrepancia de la velocidad
383 real del humanoide con V_O , menor (y más cercano a cero) es el valor arrojado por el módulo de
384 velocidad.

385 6.2.3. Dirección

386 Señala qué tan similares son la dirección objetivo (un vector unitario, que en este caso se
387 encuentra en el eje Z) y la dirección con la que camine el humanoide. Se calcula como sigue:

388

$$D = \frac{\sum_{n=0}^T e^{-C(\mathbf{v}_{t_n} \cdot \mathbf{V}_O - 1)^2}}{N} \quad (18)$$

389

390 donde t_0 es el tiempo inicial, t_T el tiempo final, \mathbf{v}_{t_n} el versor de la dirección del humanoide en
391 el momento t_n y \mathbf{V}_O el versor de la dirección objetivo.

392 El producto escalar entre los versores responde a la Similitud Coseno: $\cos \theta = \frac{\mathbf{A} \cdot \mathbf{B}}{AB}$, donde \mathbf{A} y
393 \mathbf{B} son vectores que no se encuentran normalizados, y θ es el ángulo formado entre ellos. Así, si
394 $\cos \theta = 1$, significa que los vectores están paralelos entre sí (que es el efecto buscado en el caso
395 de la dirección).

396 Al producto escalar se le resta 1, para que el módulo sea consistente con la función exponencial
397 utilizada y que valga 1 cuando $\theta = 0$, y 0 cuando $\theta = \pi$. Cabe aclarar que se trata al ángulo en
398 forma simétrica, ya que, por ejemplo $\cos(-\pi/6) = \cos(\pi/6)$.

399

400 6.2.4. Simetría

401 Señala qué tan equidistantes se encuentran los pies de la cadera, a lo largo de la caminata.
402 Aplicando solamente los módulos antes mencionados, provocaba resultados en donde una pierna
403 quedaba más distante de la pelvis que la otra, lo que provocaba que el humanoide se terminara
404 arrastrando (y posiblemente afectando a la velocidad).

405 Para mayor simplicidad, la simetría S se calculó a partir de los pies (y no de las piernas). Se
406 tomaron en cuenta sólo los ejes X y Z, porque son los relacionados a la velocidad y a la dirección,
407 respectivamente.

$$S = \frac{\sum_{n=0}^T \frac{1}{2}[e^{-C(lf_z + rf_z)^2} + e^{-C(lf_x + rf_x)^2}]}{N} \quad (19)$$

408

409 donde lf_x y lf_z es la distancia desde el pie izquierdo hasta la pelvis en los ejes X y Z, respecti-
410 vamente; y en donde rf_x y rf_z es lo mismo, pero para el pie derecho.

411

412 6.2.5. Pies abajo

413 Con los módulos señalados anteriormente, se resalta que el humanoide camine con una velo-
414 cidad y dirección determinadas, que no se caiga y que mantenga simetría mientras ejecuta sus
415 movimientos. Pero, todo esto daría, en el mejor de los casos, una caminata estilo “estrella”.
416 Sin embargo, una característica fundamental en una caminata normal es que las piernas (ergo,
417 los pies también) no sobrepasen la cadera. Si bien ésta es una propiedad negativa (expresa lo que
418 no debe tener una caminata), y se puede correr el riesgo de restringir demasiado, su ausencia da
419 resultados peores.

$$PA = \frac{\sum_{n=0}^T \frac{1}{2}(\alpha[e^{-C(lf^2)} + e^{-C(rf^2)}])}{N} \quad (20)$$

420

421 donde lf y rf son las diferencias entre la posición inicial de los pies y la altura en el momento
422 t_n de los pies izquierdo y derecho, respectivamente, y $\alpha = \max(\min(lf, rf) - hip, 0), 1$, siendo
423 lf , rf y hip las alturas del pie izquierdo, pie derecho, y la cadera (es decir, vale 0 si la altura del
424 pie izquierdo o derecho supera a la de la cadera, y 1 en otro caso).

425

426 6.3. Parámetros del Algoritmo

427 6.3.1. Métodos de selección

428 De la vasta cantidad de métodos de selección que existen, se utilizaron: ***Elite*** (en donde se
429 selecciona el individuo con mayor aptitud de la población-; y ***Roulette*** (método probabilístico,
430 que selecciona un individuo de la población total al azar, con una probabilidad proporcional a
431 su *fitness*).

432 6.3.2. Métodos de crusa

433 El método de crusa (o *crossover*) utilizado es el siguiente: De dos individuos (los padres), se
434 originan dos nuevos individuos (los hijos). Se toma cada uno de los genes de los padres y se elige,
435 con una probabilidad uniforme, uno de ellos para un hijo y el otro para el otro hijo.
436 La probabilidad de que este proceso ocurra es de 0.9.

437 6.3.3. Mutación

438 En el caso de la mutación, para cada gen del individuo, se decide con cierta probabilidad si
439 se lo muta o no. En caso afirmativo, se cambia ese gen por un valor aleatorio (que esté dentro
440 de su rango definido).

441 Dicha probabilidad de mutación es de 0.3.

442 7. Resultados Obtenidos

443 Considerando los individuos definidos en el punto xx, se realizaron experimentos, corriendo
 444 el algoritmo genético, y evaluando el resultado obtenido posteriormente (ya sea numérica o
 445 gráficamente). A continuación se analizan distintos aspectos relevantes.

446 7.1. Evolución del *fitness* según tipo de individuo

447 En el caso del individuo de tipo 2, puede observarse que, aun cuando el *fitness* llega a 0.6 (y
 448 no a 1, su cota superior), tiene un buen rendimiento, ya que, suponiendo que cada uno de los
 449 módulos del *fitness* están al 90 %, se tendría que $0.9^5 = 0.59$. Los individuos que usan actuadores
 450 de Fourier son los que producen *fitness* más bajos; el de orden 2 (individuo de tipo 3) se estanca a
 451 las pocas generaciones (al igual que el de tipo 1). El *fitness* del individuo con actuadores Fourier
 452 de orden 9 se “meseta” progresivamente (por la generación 500).

453 Por otra parte, el individuo de tipo 5 (con actuadores doble coseno), va mejorando su *fitness*
 454 paulatinamente. Esto es ben

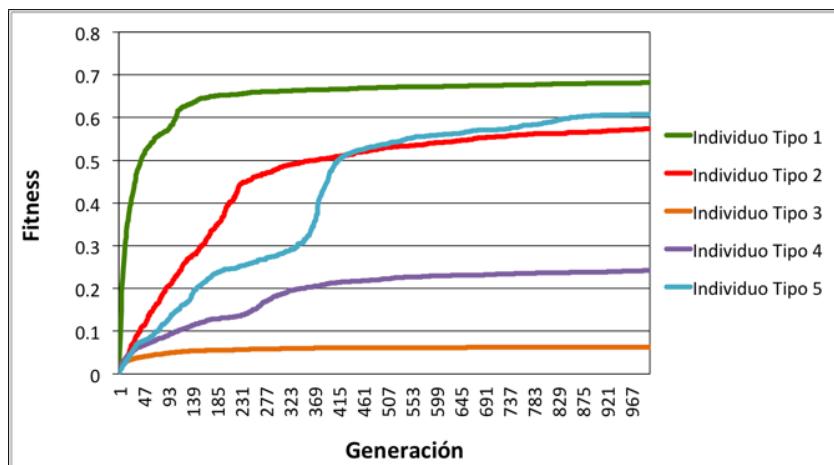


Figura 9: Evolución del *fitness* según tipo de individuo

455 7.2. Velocidad según tipo de individuo

456 A continuación, se muestra, para cada individuo, su velocidad instantánea a lo largo del
 457 tiempo.

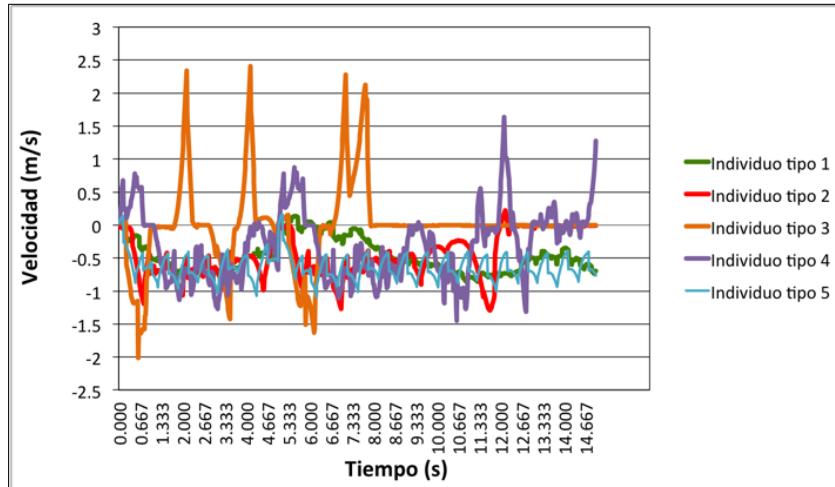


Figura 10: Velocidad según tipo de individuo

458 Como puede observarse en el caso de los individuos 3, 4 y 5 (Fourier de orden 2, Fourier
 459 de orden 9 y Doble Coseno, respectivamente), se producen picos altos y pronunciados en la
 460 velocidad. Eso repercute en que la velocidad media no sea $1.3 \frac{m}{s}$ (que es la velocidad objetivo),
 461 y ergo, en el módulo de velocidad del *fitness* (provocando que éste sea más bajo).
 462 En el caso de los individuos 1 y 2, la velocidad instantánea oscila de forma suave.

463 7.3. Altura según tipo de individuo

464 Como puede identificarse en la figura 11, la altura de los individuos es otra característica para
 465 diferenciarlos en su rendimiento.
 466 En efecto, los individuos de tipo 3 y 4 (que utilizan actuadores de Fourier de orden 2 y 9,
 467 respectivamente), son los que caen de forma más abrupta, y aunque intentan levantarse, vuelven
 468 a caer con la misma intensidad.
 469 Por su parte, el individuo de tipo 1 es el que mantiene la altura constante, a causa del riel que
 470 tiene activado (que le hace mantener la altura). El individuo de tipo 2, con actuadores genéricos
 471 y función partida, logra mantener la altura, hasta que cae. Considerando que no tiene el riel
 472 activado, y que cuando cae no vuelve a levantarse (ya que en ese movimiento suelen emplearse
 473 los brazos y manos), tiene un comportamiento similar a una caminata real.
 474 En lo que respecta al individuo de tipo 5, que utiliza como actuadores la función de doble coseno,
 475 se cae y levanta repetidamente, pero de forma más suave que los que utilizan Fourier de orden 2
 476 y 9.

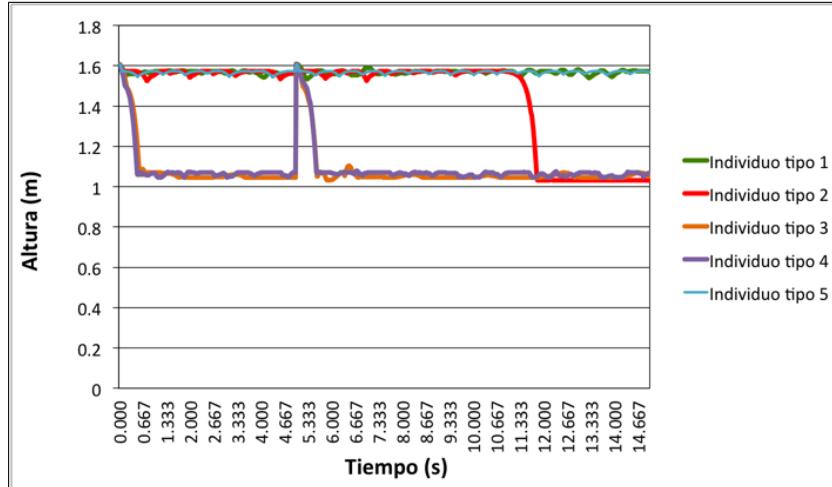


Figura 11: Altura (posición de la pelvis) según tipo de individuo

7.4. Comparación de tipos de individuo

Principalmente el generico y el doble coseno son los que mejor andan, tanto en fitness como en altura.

Pero son dos caminatas distintas, una un poco más natural (generico) y la otra más robotica" pero estable.

Si hay que elegir como mejor a uno de los dos, habria que decidir entre realismo (el humanoide se cae, pero es más natural), y estabilidad (no se cae nunca, pero parece un "passive walker").

La funcion de fitness que se plantea (con sus módulos) indica propiedades ideales para una caminata, pero que no implican realismo en la misma. Y entonces dos individuos con fitness parecidos dieron como fruto caminatas muy distintas.

Los actuadores fueron importantísimos.

7.5. Video

Acá iría el link del video?

8. Conclusiones

xxxxxxxxxx

Referencias

- [1] Wikipedia: https://es.wikipedia.org/wiki/Physics_engine
- [2] Andreas Gerndt y otros, *An Evaluation of Open Source Physics Engines for Use in Virtual Reality Assembly Simulations*. Fecha de publicación: 2012
- [3] Tom Erez y otros, *Simulation Tools for Model-Based Robotics: Comparison of Bullet, Havok, MuJoCo, ODE and PhysX*
- [4] Sitio web de Bullet Physics: <http://www.bulletphysics.org/>

- 499 [5] Erin Catto, *Iterative Dynamic with Temporal Coherence*. Fecha de publicación: 2005
- 500 [6] Sitio web de GaLib: <http://lancet.mit.edu/ga/>
- 501 [7] Thomas Geijtenbeek, Michiel van de Panne y A. Frank van der Stappen, *Flexible Muscle-*
502 *Based Locomotion for Bipedal Creatures*, 2013
- 503 [8] Marek Wojtyra, *Multibody Simulation Model of Human Walking - Warsaw University of*
504 *Technology, 2003*
- 505 [9] <http://www.exrx.net/Kinesiology/Segments.html>
- 506 [10] Kevin Kenny, Máximo Videla y Axel Wassington, *Proyecto Final para la obtención del título:*
507 *Ingeniero en Informática - ITBA, 2014*