

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](mailto:ealtamir@itba.edu.ar)

10

**Teresa Fontanella De Santis**

11

[tfontane@itba.edu.ar](mailto:tfontane@itba.edu.ar)

12

**Tomás Mehdi**

13

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

58	6.3.1	Métodos de selección . . . . .	20
59	6.3.2	Métodos de crusa . . . . .	20
60	6.3.3	Mutación . . . . .	20
61	<b>7</b>	<b>Resultados Obtenidos</b>	<b>20</b>
62	7.1	Evolución del <i>fitness</i> según tipo de individuo . . . . .	20
63	7.2	Velocidad según tipo de individuo . . . . .	21
64	7.3	Altura según tipo de individuo . . . . .	21
65	7.4	Comparación de tipos de individuo . . . . .	22
66	7.5	Video . . . . .	22
67	<b>8</b>	<b>Conclusiones</b>	<b>22</b>
68	<b>Referencias</b>		<b>22</b>

69

## Resumen

70

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

73

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

78

## 1. Introducción

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

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

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

89 Este problema se puede abordar de diversas maneras, involucrando: redes neuronales<sup>1</sup>, sistemas  
90 de control (*passive walkers*<sup>2</sup>), algoritmos genéticos, entre otras. Pero varios de ellos implican  
91 modelos teóricos complejos de humanoide (considerando músculos con distintos materiales, etc.).

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

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

---

<sup>1</sup>Una 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.

<sup>2</sup>Un *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).

## <sup>99</sup> 2. Herramientas

### <sup>100</sup> 2.1. Motor Físico

<sup>101</sup> Se le llama motor físico o *physics engine* a un “*software* capaz de realizar simulaciones de  
<sup>102</sup> ciertos sistemas físicos, como la dinámica del cuerpo rígido, el movimiento de un fluido y la  
<sup>103</sup> elasticidad”[1].

<sup>104</sup> Actualmente, existen muchos motores físicos: ya sea de código propietario (PhysX, Havok), como  
<sup>105</sup> *open-source* (*Bullet Physics*, *Box2D*, Newton, OGRE). Considerando análisis relacionados [2][3],  
<sup>106</sup> y la necesidad de que el espacio simulado fuese en 3D, se decidió que *Bullet Physics*[4] es el más  
<sup>107</sup> idóneo. Está implementado en C++ y ha sido utilizado en varios juegos (*Grand Theft Auto IV* y  
<sup>108</sup> V, etc); en los efectos especiales de películas (Hancock, Bolt, etc.); y proyectos científicos, como  
<sup>109</sup> la herramienta *open-source Tensegrity Robotics Toolkit* de la NASA<sup>3</sup>; entre otros.

<sup>110</sup> Si bien (como se verá más adelante) *Bullet* tiene problemas asociados con el coeficiente de restitu-  
<sup>111</sup> ción, posee una muy buena *performance* en la detección de colisiones, la dinámica y la resolución  
<sup>112</sup> de *constraints*. Esto se debe, en parte, a diferentes algoritmos iterativos de orden lineal (donde  
<sup>113</sup> el más importante es *Sequential Impulse*), de *caching* y también a la utilización de un modelo  
<sup>114</sup> de fricción de Coulomb aproximado [5]. Además, el motor físico brinda la posibilidad de regular  
<sup>115</sup> la precisión requerida en estos cálculos (sin olvidar que, con iguales recursos, a mayor precisión,  
<sup>116</sup> mayor capacidad de cómputo requerida y, ergo, mayor tiempo). Dado que la construcción del hu-  
<sup>117</sup> manoide implica definir características y restricciones de movimiento de cada una de sus partes,  
<sup>118</sup> lo antes mencionado fue crucial para la elección de *Bullet Physics* en este proyecto.

#### <sup>119</sup> 2.1.1. Funcionamiento

<sup>120</sup> El motor físico se encarga de la simulación de cuerpos rígidos y la interacción entre los mismos.  
<sup>121</sup> En particular debe: calcular el resultado de colisiones, arreglar el solapamiento de los cuerpos  
<sup>122</sup> en el espacio de simulación, estimar las fuerzas producidas debido a la fricción, y mantener el  
<sup>123</sup> cumplimiento de restricciones que puedan existir entre los cuerpos (por ejemplo, un vínculo para  
<sup>124</sup> formar una articulación). Para lograrlo, *Bullet* modela, a partir de un conjunto de ecuaciones,  
<sup>125</sup> las distintas restricciones que deben ser respetadas.

<sup>126</sup> Estos modelos reciben como dato la velocidad lineal y angular de cada objeto, y las fuerzas  
<sup>127</sup> que actúan sobre los mismos. Dada esta información, se resuelve el sistema de ecuaciones, cuya  
<sup>128</sup> solución representa las magnitudes de las fuerzas a accionar sobre el mismo, a fin de satisfacer  
<sup>129</sup> todas las restricciones. Para encontrar esta solución, entran en juego los distintos métodos de  
<sup>130</sup> complejidad lineal mencionados en la subsección anterior.

<sup>131</sup> Este procedimiento, se lleva a cabo en cada *timestep* de la simulación; donde un *timestep* es el  
<sup>132</sup> intervalo de tiempo que transcurre entre un cálculo de magnitudes y otro. Mientras menor sea  
<sup>133</sup> el *timestep*, el simulador será más preciso pero también consumirá más tiempo de cómputo.

#### <sup>134</sup> 2.1.2. Modelo de fricción utilizado y su verificación

<sup>135</sup> Hay reglas físicas relacionadas con el entorno y que son muy importantes para la caminata:  
<sup>136</sup> el modelo de fricción, con sus respectivos coeficientes de fricción y restitución.

<sup>137</sup>  
<sup>138</sup>

---

<sup>3</sup><http://bulletphysics.org/Bullet/phpBB3/viewtopic.php?f=17&t=9978>

139 En base a los modelos físico-matemáticos utilizados en los dos fenómenos en cuestión (y que  
 140 se explicarán a continuación), y pensando en posibles futuras simulaciones de varios humanoides  
 141 chocando e interactuando entre sí; se llevaron a cabo dos experimentos para verificar que  
 142 estuvieran en concordancia con los datos arrojados por *Bullet*.

143 **2.1.2.1. Verificación del coeficiente de fricción**

144 El primero simula un cubo, de  $m_{cube} = 1kg$  y  $l_{box} = 1m$ , que tiene una velocidad inicial  
 145 constante ( $v_i$ ) en el eje horizontal, que gradualmente se detiene por acción de la fricción, hasta  
 146 llegar al reposo (Fig. 1). Se buscó determinar el modelo utilizado por *Bullet* para simular las  
 147 fuerzas resultantes sobre un cuerpo por acción de la fricción.

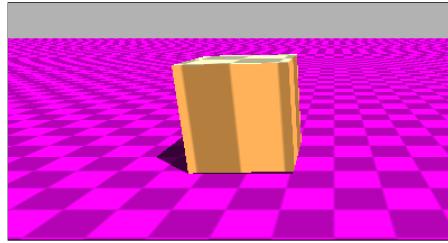


Figura 1: RENOMBRAR FIGURA

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

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

150  
 151 En este caso, el cuerpo empieza su movimiento en el origen, por lo tanto la posición inicial ( $x_i$ )  
 152 es cero. Debido a la fricción entre el cuerpo y el suelo, se genera una fuerza de rozamiento  $F_{\mu_d}$   
 153 (ec. (2)) en la misma dirección que la velocidad del sólido y en sentido contrario.

154

$$(F_{\mu_d} = \mu_d F_N) \quad (2)$$

155 donde  $F_N = mg$  es la fuerza normal que actúa sobre la caja por acción de la gravedad  $g$ , y  $\mu_d$  es  
 156 el coeficiente de fricción dinámico.

157

158 Finalmente, se obtiene la aceleración:

$$a = \frac{F_{\mu_d}}{m} = \frac{-\mu_d F_N}{m} = \frac{-\mu_d mg}{m} = (\mu_d mg) \quad (3)$$

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

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

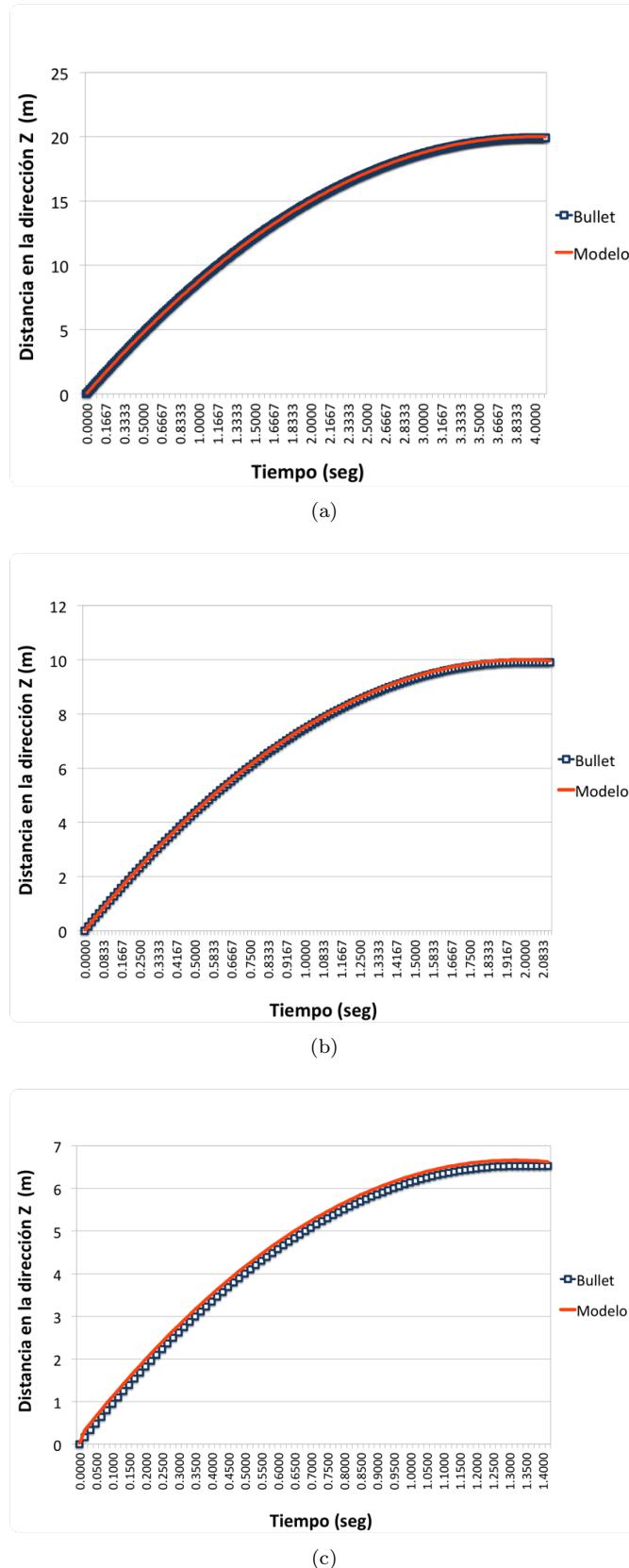


Figura 2: Resultados logrados de simular el sistema descripto en Fig. 1, usando  $v_i = 10 \frac{m}{s}$  y:  
 (a)  $\mu_d = 0.25$ , (b)  $\mu_d = 0.50$ , y (c)  $\mu_d = 0.75$

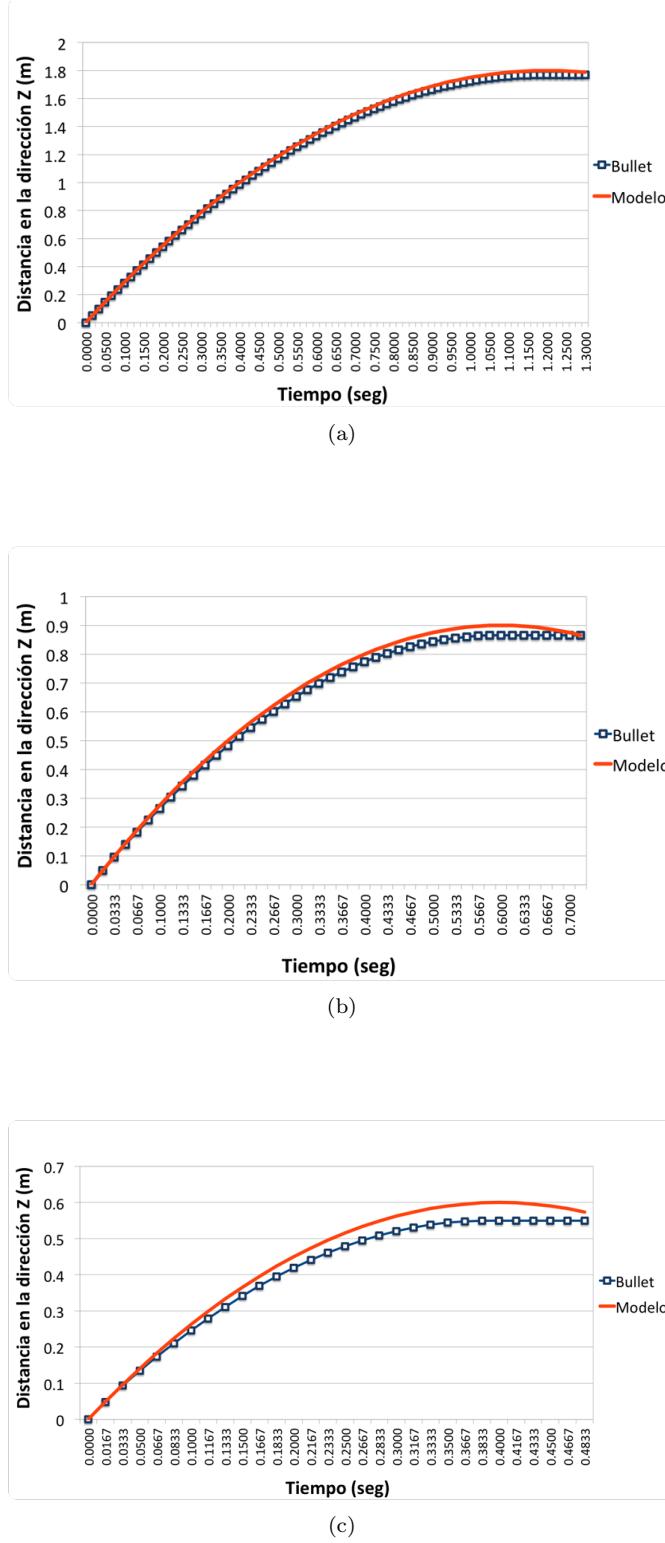


Figura 3: Resultados logrados de simular el sistema descripto en Fig. 1, usando  $v_i = 3 \frac{m}{s}$  y:  
 (a)  $\mu_d = 0.25$ , (b)  $\mu_d = 0.50$ , y (c)  $\mu_d = 0.75$

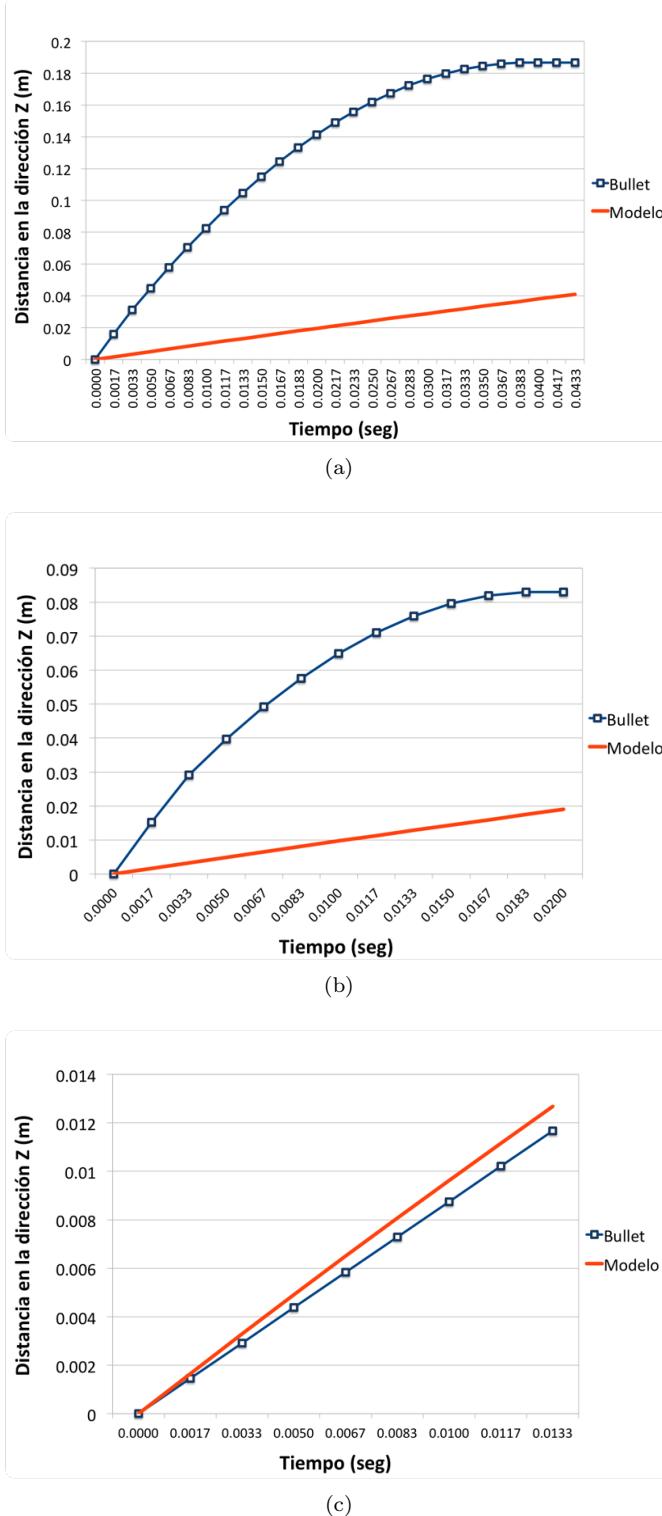


Figura 4: Resultados logrados de simular el sistema descripto en Fig. 1, usando  $v_i = 1 \frac{m}{s}$  y:

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

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

165 Cabe destacar que los gráficos que corresponden a la Fig. 4, presentan una discrepancia mayor  
 166 entre la simulación y el modelo. Este hecho puede deberse a que dicha prueba tiene una cantidad  
 167 relativamente baja de puntos, por lo tanto la resolución de la simulación no es lo suficientemente  
 168 buena y el simulador tiende a cometer un mayor error.

169 **2.1.2.2. Verificación del coeficiente de restitución**

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

174

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

175

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

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

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

185

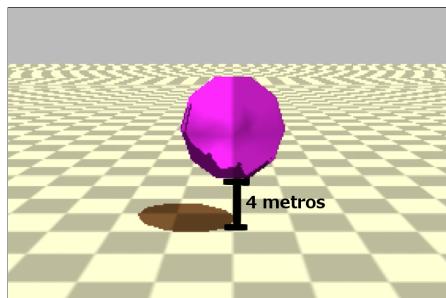


Figura 5: RENOMBRAR FIGURA

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

190

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

191 Se muestran los experimentos numéricos realizados. En ellos,  $v_i = \{-0.5, -3.5, -4, -5 \text{ y } -10\} \frac{\text{m}}{\text{s}}$   
 192 y  $e_{sim} = \{0.2, 0.5 \text{ y } 0.8\}$ .

$v_i$	$e_{sim}$	$v_f$	$e_{medida}$	$\epsilon_{rel}$
$-0.5 \frac{m}{s}$	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 descripto en Fig. 5

193 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  
 194 evidencia en la Tabla 1. En cada una de ellas el error fue de casi 1. La razón por la que ocurre este  
 195 hecho se debe a que *Bullet* utiliza un algoritmo de colisión que frena la velocidad de un objeto  
 196 que está a punto de colisionar. Haciendo esto puede evitar que los sólidos se traspasen y de esta  
 197 forma se pueden realizar cálculos de fuerza más precisos.  
 198

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

### 204 2.1.3. Ventajas

205 Las ventajas del motor físico son:

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

### 209 2.1.4. Desventajas

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

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

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

217 **2.2. Librería de Algoritmos Genéticos**

218 Se utilizó la conocida librería de Algoritmos Genéticos para C++ GaLib, desarrollada por  
219 Matthew Wall del MIT [7].  
220 Ofrece funcionalidades como: programación paralela, diversos métodos de selección (*elite*, *rou-  
lette*), estrategias de reemplazo (de padres, aleatorio, del peor), entre otras.

222 **2.3. Código Fuente**

223 Al estar *Bullet* implementado en C++, el código fuente también está desarrollado en ese  
224 lenguaje. En *Bullet*, se define un *World* (o mundo físico) en donde se puede insertar, entre otras  
225 cosas, cuerpos rígidos. En este caso en particular, el mundo consta de un plano (el suelo) y el  
226 humanoide encima (compuesto por cuerpos rígidos y otros elementos físicos).  
227 El *software* creado incluye: creación del humanoide, siendo éste capaz de desplazarse por medio  
228 de actuadores (que se verán en la Sección 4); el desarrollo del algoritmo genético (la definición de  
229 los individuos, la función de *fitness*, métodos de selección, etc.); visualización gráfica del mejor  
230 humanoide logrado por el algoritmo genético; y la posibilidad de realizar gráficos referidos a la  
231 evolución del algoritmo genético (*fitness* por cada generación, etc.).

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

233 **3. Modelo Utilizado**

234 Hay diversos modelos. Algunos son más genéricos [8] [9] y complejos. Sin embargo, se procuró  
235 utilizar uno que fuera sencillo pero representativo a la vez.  
236 Se modela al cuerpo humano, con el motor *Bullet Physics*, como un conjunto de segmentos unidos  
237 por articulaciones (Fig. 6). A cada uno de ellos se les aplica una fuerza en el centro de masa  
238 de cada segmento (denominada Actuador). Que la caminata se produzca o no, depende del tipo  
239 de actuador utilizado (la función utilizada para la fuerza), y de sus parámetros. El objetivo,  
240 entonces, se reduce a encontrar dichos parámetros. Para eso se usan los algoritmos genéticos, un  
241 método de Inteligencia Artificial. De este modo, se obtiene, de forma análoga a la selección natu-  
242 ral, los individuos que mejor se adapten a la caminata. Tanto los actuadores como el algoritmo  
243 genético se explicarán más adelante.

244

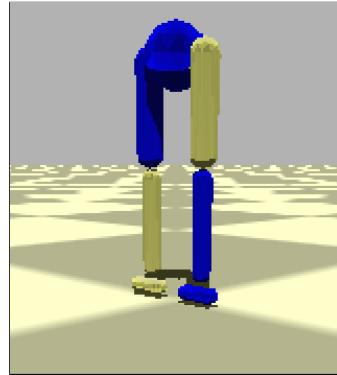


Figura 6: Humanoide diseñado

### **245 3.1. Composición Física del Humanoide**

246 Como ya se expresó, el humanoide fue modelado en *Bullet* como un conjunto de segmentos  
 247 (cuerpos rígidos), unidos por articulaciones (ver Fig. 7).

248 Se dividió al cuerpo humano en: cabeza, tronco, miembro superior, pelvis y miembro inferior  
 249 (muslo, pierna y pie). Considerar la mitad superior del cuerpo implicaba mayor complejidad  
 250 (manejo de equilibrio, humanoide más pesado y con mayor volumen, etc.). Ergo, solo se tomó la  
 251 pelvis y el miembro inferior.

252 A continuación se presenta la composición de cada segmento (de acuerdo a la biomecánica [10]):

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

### **253 3.2. Articulaciones**

254 Para unir los distintos segmentos entre sí, se utilizaron articulaciones bisagra con 1 grado de  
 255 libertad: en el eje Z, donde ocurre la caminata; y en el eje Y, perpendicular al piso (ver Fig. 7).  
 256 Asimismo, para cada caso en particular, se definieron cotas para los ángulos que pueden existir  
 257 entre los segmentos. Esto es muy importante, no sólo porque se adecúa a datos biológicos, sino  
 258 porque, de otro modo la caminata no podría lograrse: si los ángulos son demasiado altos, la  
 259 caminata se produce girando las piernas por encima de la pelvis; si por el contrario, son demasiado  
 260 bajos, las piernas van a estar muy rígidas, originando pocos pasos y muy cortos.

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

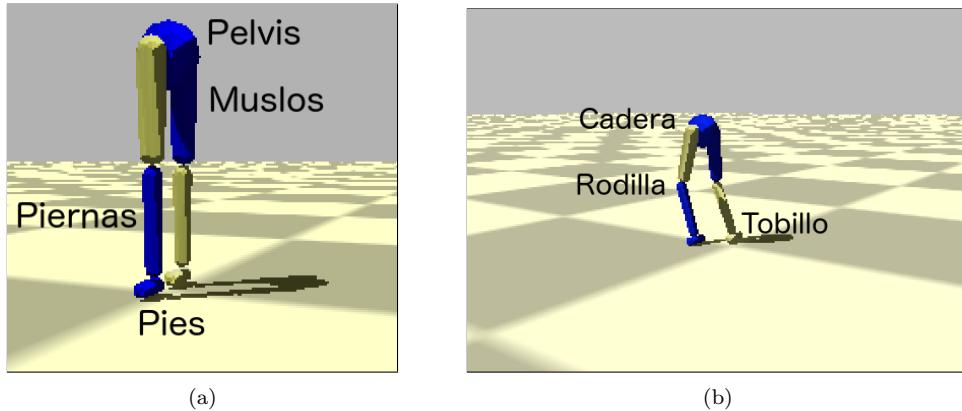


Figura 7: Humanoide diseñado: (a) segmentos, y (b) articulaciones

## 4. Actuadores

A cada uno de los segmentos correspondientes al muslo y la pierna del bípedo, se le aplica un torque (o actuador) en el eje X (perpendicular a la trayectoria), como se ve en Fig 8. Así, pueden moverse para arriba o para abajo (con respecto a la articulación a la que pertenecen). A fin de simplificar el modelo, el humanoide tiene el mismo tipo de actuador utilizado en todos los segmentos. Que el torque se aplique en una sola dimensión, contribuye a que la caminata producida sea plana (en 2D, y no en 3D, como debería ser en una caminata real). Para indicar el módulo de dicho torque, se diseñaron diferentes funciones (todas ellas periódicas), mencionadas en las subsecciones que siguen.

272

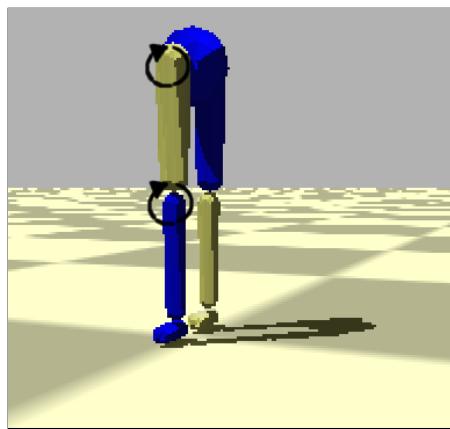


Figura 8: Aplicación de los actuadores en los segmentos del bípedo

### 4.1. Genérico

273 Es el actuador más sencillo, tanto matemática como computacionalmente. La fase es la misma tanto en el seno como en el coseno, para evitar que se formen otro tipo de funciones no cíclicas.

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

<sup>276</sup> **4.2. Fourier**

<sup>277</sup> Este actuador utiliza una serie de Fourier de dos términos.

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

<sup>278</sup> **4.3. Extra Fourier**

<sup>279</sup> Es una extensión del actuador anterior, pero con 9 términos. Por ser de mayor grado, brinda  
<sup>280</sup> una mayor precisión. Sin embargo, es más difícil de manejar computacionalmente; y, además,  
<sup>281</sup> que sea más preciso no garantiza que con él se pueda lograr una buena caminata.

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

<sup>282</sup> **4.4. Doble coseno**

<sup>283</sup> Esta función periódica utiliza medio ciclo de una función sinusoidal, y medio ciclo de otra  
<sup>284</sup> (ambas pueden tener frecuencias distintas). De esta manera, se consigue una caminata más  
<sup>285</sup> natural, y que no ocurre con los actuadores de Fourier, que producen una doble flexión de las  
<sup>286</sup> rodillas en cada ciclo.

<sup>287</sup> La idea es lograr una función periódica a partir de una que no lo es (ya que  $t$  es lineal). Para  
<sup>288</sup> eso, se utiliza  $\psi(t)$  (ec. (10)) que aplica una transformación a los números reales, para que se  
<sup>289</sup> encuentren dentro del rango del ciclo completo (con las dos frecuencias).  $\omega$  es la frecuencia de  
<sup>290</sup>  $f(t)$  (ec. (12)), que utiliza medio ciclo con frecuencia  $\omega_1$  y medio ciclo con frecuencia  $\omega_2$ .

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

<sup>291</sup>

<sup>292</sup>

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

<sup>293</sup>

<sup>294</sup>

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

<sup>295</sup> **5. Condiciones iniciales y de contorno**

<sup>296</sup> Como las funciones periódicas señaladas en los actuadores, no son suficientes para lograr la  
<sup>297</sup> caminata, se le adosaron las funciones que se detallan seguidamente.

<sup>298</sup> **5.1. Función Partida**

<sup>299</sup> El andar del humanoide es cíclico. Sin embargo, por la posición inicial del individuo, se requiere  
<sup>300</sup> para el tiempo del primer paso, una función distinta a la del resto de la caminata. El tipo de

301 función puede ser cualquiera de los actuadores vistos anteriormente (pero no necesariamente con  
 302 los mismos valores de amplitud, frecuencia y fase asignados a las piernas). Empero, se utilizó la  
 303 función vista en el actuador genérico.

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

## 306 5.2. Fase sincronizada

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

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

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

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

## 313 6. Algoritmo Genético

### 314 6.1. Individuo

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

#### 317 6.1.1. Parámetros

318 Tanto la función partida, como los actuadores, tienen como parámetros: amplitud ( $A$  o  $B$ );  
 319 fase ( $\phi$ ), que indica dónde comienza el paso y se mide en radianes; frecuencia ( $\omega$ ); y término  
 320 independiente ( $C$ ).

#### 321 6.1.2. Valores

322 Cada uno de los segmentos tiene propiedades físicas distintas (masa, largo, etc.), razón por  
 323 la cual no necesariamente sus genes deban tener los mismos rangos de valores.

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	$-\pi$	$\pi$
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	$-\pi$	$\pi$
	Muslo y Pierna	Término independiente	-10	10
Fourier	Muslo	Amplitud	-60	60
	Pierna	Amplitud	-30	30
	Muslo y Pierna	Frecuencia	0.01	10
	Muslo y Pierna	Fase	$-\pi$	$\pi$
	Muslo y Pierna	Término independiente	-10	10
Extra Fourier	Muslo y Pierna	Amplitud	-60	60
	Muslo y Pierna	Frecuencia	0.1	2
	Muslo y Pierna	Fase	$-\pi$	$\pi$
	Muslo y Pierna	Término independiente	-10	10
Doble Coseno	Muslo y Pierna	Amplitud	-100	100
	Muslo y Pierna	Frecuencia	0.01	10
	Muslo y Pierna	Fase	$-\pi$	$\pi$
	Muslo	Término independiente	-50	50
	Pierna	Término independiente	-100	100

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

#### 324 6.1.3. Implementaciones de individuos

325 Para favorecer el análisis de las distintas características antes implementadas, se implemen-  
326 taron los siguientes individuos:

Tipo de Individuo	Actuador	Función Partida	Fase sincronizada
1	Genérico	No	Sí
2	Genérico	Sí	Sí
3	Fourier	Sí	Sí
4	Extra Fourier	Sí	Sí
5	Doble coseno	Sí	Sí

Tabla 5: Tipos de humanoide

#### 327 6.1.4. Constitución del cromosoma

328 En la Tabla 6 se presenta la composición del cromosoma de cada individuo.

329 Vale aclarar que la función partida se especifica para cada segmento (las dos piernas y los dos  
330 muslos). En cambio, para los actuadores, sólo se definen dos (uno para las piernas y otro para  
331 los muslos).

Individuo	Tipo 1	Tipo 2	Tipo 3	Tipo 4	Tipo 5
<b>Amplitud</b>	12	12	16	44	8
<b>Frecuencia</b>	2	6	6	6	8
<b>Fase</b>	2	6	6	6	6
<b>Término independiente</b>	2	6	6	6	6
<b>Longitud del cromosoma</b>	18	30	34	62	28

Tabla 6: Cantidad de parámetros según tipo de individuo

## 332 6.2. Fitness

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

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

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

342

### 343 6.2.1. Altura

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

$$345 H = \frac{\sum_{n=0}^T e^{-C(h_{t_n} - h_{t_0})^2}}{N} \quad (16)$$

346

347 donde  $t_0$  es el tiempo inicial,  $t_T$  el tiempo final,  $N$  la cantidad pasos de simulación y  $C$  una  
 348 constante  $C = 5$ .

349 Se calcula a partir de la diferencia entre la altura en cada instante de la simulación, con su altura  
 350 inicial (la altura está definida como la posición de la pelvis en el eje Z). Cuanto mayor sea esa  
 351 diferencia, más rápido el individuo cae, y por eso este módulo tiende a cero. Por el contrario,  
 352 valdrá uno si la diferencia es ínfima (lo que significa que el humanoide mantiene su misma altura  
 353 durante la caminata).

354

### 355 6.2.2. Velocidad

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

$$358 V = \frac{\sum_{n=0}^T e^{-C(\|v_{t_n}\| - V_O)^2}}{N} \quad (17)$$

359

360 donde  $t_0$  es el tiempo inicial,  $t_T$  el tiempo final y  $V_O$  la velocidad objetivo en el eje Z (el eje de

361 la caminata).

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

### 365 6.2.3. Dirección

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

$$368 D = \frac{\sum_{n=0}^T e^{-C(v_{t_n} \cdot V_O - 1)^2}}{N} \quad (18)$$

369 donde  $t_0$  es el tiempo inicial,  $t_T$  el tiempo final,  $v_{t_n}$  el versor de la dirección del humanoide en el  
370 momento  $t_n$  y  $V_O$  el versor de la dirección objetivo.

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

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

379

### 380 6.2.4. Simetría

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

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

$$388 S = \frac{\sum_{n=0}^T \frac{1}{2}[e^{-C(lf_z + rf_z)^2} + e^{-C(lf_x + rf_x)^2}]}{N} \quad (19)$$

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

391

### 392 6.2.5. Pies abajo

393 Con los módulos sealados anteriormente, se resalta que el humanoide camine con una velo-  
394 cidad y dirección determinadas, que no se caiga y que mantenga simetría mientras ejecuta sus  
395 movimientos. Pero, todo esto daría, en el mejor de los casos, una caminata estilo “estrella”.

396 Sin embargo, una característica fundamental en una caminata normal es que las piernas (ergo,  
397 los pies también) no sobrepasen la cadera. Si bien ésta es una propiedad negativa (expresa lo que

398 no debe tener una caminata), y se puede correr el riesgo de restringir demasiado, su ausencia da  
 399 resultados peores.

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

400  
 401 donde  $\alpha = \max(\min(lf, rf) - hip, 0), 1$ ) (es decir, vale 0 si la altura del pie izquierdo o derecho  
 402 supera a la de la cadera, y 1 en otro caso); y  $ldf$  y  $rdf$  es la diferencia entre la posición inicial de  
 403 los pies y la altura en el momento  $t_n$  de los pies izquierdo y derecho, respectivamente.

404

### 405 6.3. Parámetros del Algoritmo

#### 406 6.3.1. Métodos de selección

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

#### 411 6.3.2. Métodos de crusa

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

#### 416 6.3.3. Mutación

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

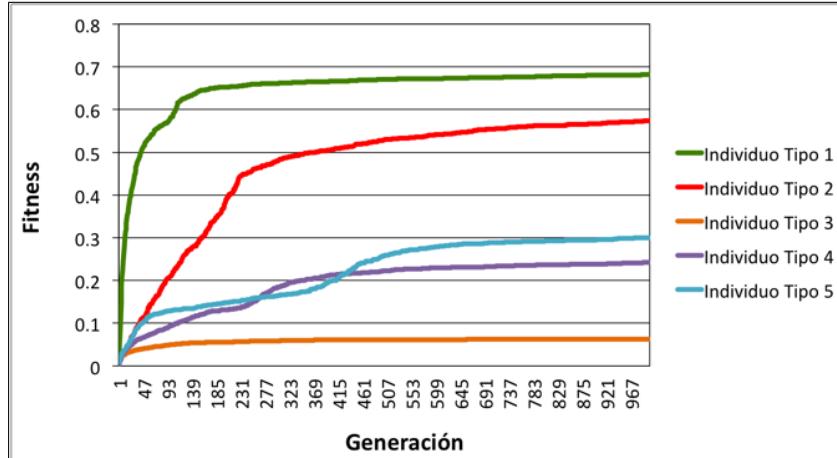
420 Dicha probabilidad de mutación es de 0.3.

## 421 7. Resultados Obtenidos

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

### 425 7.1. Evolución del *fitness* según tipo de individuo

426 En el caso del individuo de tipo 2, puede observarse que, aun cuando el *fitness* llega a 0.6 (y  
 427 no a 1, su cota superior), tiene un buen rendimiento, ya que, suponiendo que cada uno de los  
 428 módulos del *fitness* están al 90 %, se tendría que  $0.9^5 = 0.59$ .

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

## 429 7.2. Velocidad según tipo de individuo

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

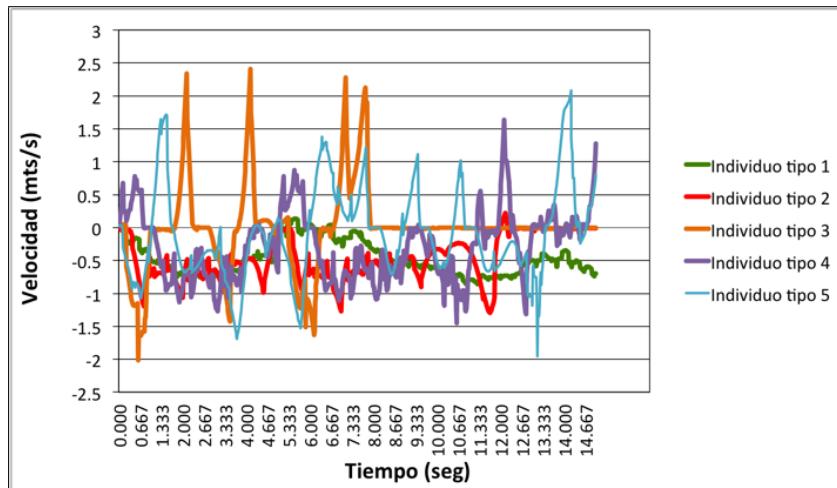


Figura 10: Velocidad según tipo de individuo

432 Como puede observarse en el caso de los individuos 3, 4 y 5 (Fourier, Extra Fourier y Doble  
 433 Coseno, respectivamente), se producen picos altos y pronunciados en la velocidad. Eso repercute  
 434 en que la velocidad media no sea  $1.3 \frac{m}{s}$  (que es la velocidad objetivo), y ergo, en el módulo de  
 435 velocidad del *fitness* (provocando que éste sea más bajo).

436 En el caso de los individuos 1 y 2, la velocidad instantánea oscila de forma suave.

## 437 7.3. Altura según tipo de individuo

438 Como puede identificarse en la figura 11, la altura de los individuos es otra característica para  
 439 diferenciarlos en su rendimiento.

440 En efecto, los individuos de tipo 3 y 4 (que utilizan actuadores de Fourier y Extra Fourier,  
 441 respectivamente), son los que caen de forma más abrupta, y aunque intentan levantarse, vuelven  
 442 a caer con la misma intensidad.

443 Por su parte, el individuo de tipo 1 es el que mantiene la altura constante, a causa del riel que  
 444 tiene activado (que le hace mantener la altura). El individuo de tipo 2, con actuadores genéricos  
 445 y función partida, logra mantener la altura, hasta que cae. Considerando que no tiene el riel  
 446 activado, y que cuando cae no vuelve a levantarse (ya que en ese movimiento suelen emplearse  
 447 los brazos y manos), tiene un comportamiento similar a una caminata real.  
 448 En lo que respecta al individuo de tipo 5, que utiliza como actuadores la función de doble coseno,  
 449 se cae y levanta repetidamente, pero de forma más suave que los que utilizan Fourier y Extra  
 450 Fourier.

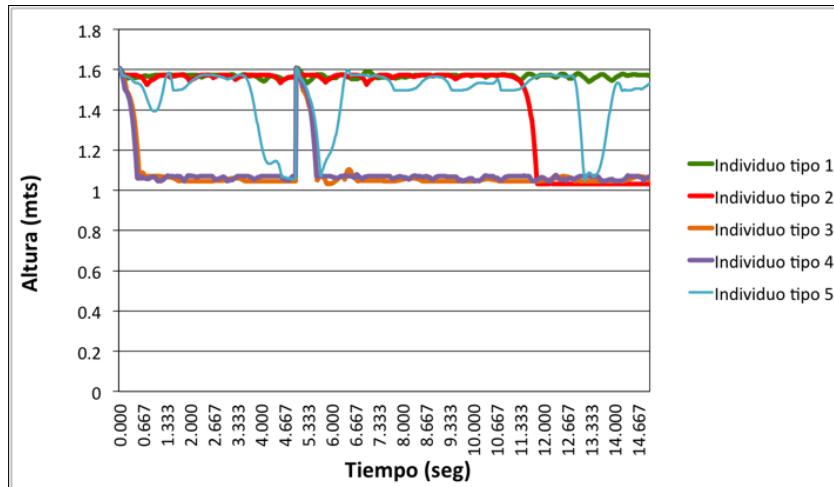


Figura 11: Altura (posición de la pelvis) según tipo de individuo

#### 451 7.4. Comparación de tipos de individuo

#### 452 7.5. Video

453 Acá iría el link del video?

## 454 8. Conclusiones

455 XXXXXXXX

## 456 Referencias

- 457 [1] Wikipedia: [https://es.wikipedia.org/wiki/Physics\\_engine](https://es.wikipedia.org/wiki/Physics_engine)
- 458 [2] Andreas Gerndt y otros, *An Evaluation of Open Source Physics Engines for Use in Virtual*  
 459 *Reality Assembly Simulations*. Fecha de publicación: 2012
- 460 [3] Tom Erez y otros, *Simulation Tools for Model-Based Robotics: Comparison of Bullet, Havok,*  
 461 *MuJoCo, ODE and PhysX*
- 462 [4] Sitio web de Bullet Physics: <http://www.bulletphysics.org/>
- 463 [5] Erin Catto, *Iterative Dynamic with Temporal Coherence*. Fecha de publicación: 2005

- 464 [6] Kevin Kenny, Máximo Videla y Axel Wassington, *Proyecto Final para la obtención del título:*  
465       *Ingeniero en Informática - ITBA, 2014*
- 466 [7] Sitio web de GaLib: <http://lancet.mit.edu/ga/>
- 467 [8] Thomas Geijtenbeek, Michiel van de Panne y A. Frank van der Stappen, *Flexible Muscle-*  
468       *Based Locomotion for Bipedal Creatures, 2013*
- 469 [9] Marek Wojtyra, *Multibody Simulation Model of Human Walking - Warsaw University of*  
470       *Technology, 2003*
- 471 [10] <http://www.exrx.net/Kinesiology/Segments.html>