

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

**Abril 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 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	<b>3 Modelo utilizado</b>	12
33	3.1 Composición física del humanoide . . . . .	12
34	3.2 Articulaciones . . . . .	13
35	<b>4 Actuadores</b>	14
36	4.1 Genérico . . . . .	14
37	4.2 Coseno doble frecuencia . . . . .	15
38	4.3 Fourier de orden 2 . . . . .	16
39	4.4 Fourier de orden 9 . . . . .	16
40	<b>5 Condiciones iniciales y de contorno</b>	17
41	5.1 Función partida . . . . .	17
42	5.2 Fase sincronizada . . . . .	17
43	<b>6 Algoritmo genético</b>	17
44	6.1 Individuo . . . . .	17
45	6.1.1 Parámetros . . . . .	18
46	6.1.2 Valores . . . . .	18
47	6.1.3 Implementación de individuos . . . . .	19
48	6.1.4 Constitución del cromosoma . . . . .	19
49	6.2 Fitness . . . . .	20
50	6.2.1 Altura . . . . .	20
51	6.2.2 Velocidad . . . . .	21
52	6.2.3 Dirección . . . . .	21
53	6.2.4 Simetría . . . . .	21
54	6.2.5 Pies abajo . . . . .	22
55	6.3 Operadores del algoritmo . . . . .	22
56	6.3.1 Métodos de selección . . . . .	22
57	6.3.2 Métodos de cruce . . . . .	22
58	6.3.3 Mutación . . . . .	22

59	6.3.4 Otras características . . . . .	23
60	<b>7 Resultados obtenidos</b>	<b>23</b>
61	7.1 Evolución del <i>fitness</i> según tipo de individuos . . . . .	24
62	7.2 Velocidad según tipo de individuos . . . . .	24
63	7.3 Altura según tipo de individuos . . . . .	25
64	7.4 Comparación de tipo de individuos . . . . .	25
65	7.5 Video . . . . .	26
66	<b>8 Conclusiones</b>	<b>26</b>
67	<b>Referencias</b>	<b>27</b>

68

## Resumen

69

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

- 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  
74 reales y exactos.
- 75 – Inteligencia artificial: que aprenda a caminar por sí mismo, utilizando para ello  
76 métodos de *soft computing* como algoritmos genéticos.

## 77 1. Introducción

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

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

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

88 Este problema se puede abordar de diversas maneras, involucrando: redes neuronales<sup>1</sup>, sistemas  
89 de control<sup>2</sup> (*passive walkers*<sup>3</sup>)[1], algoritmos genéticos<sup>4</sup>[2], entre otras. Varios de ellos implican  
90 modelos teóricos complejos de humanoide (considerando músculos con distintos materiales, etc.).

91 Asimismo, existe un trabajo previo cuyo propósito era lograr la caminata de un cuadrúpedo  
92 virtual, utilizando algoritmos genéticos[3]. Continuando con esta última línea de investigación y  
93 extendiéndola a bípedos, en este proyecto también se buscó aplicar algoritmos genéticos, y lograr  
94 la caminata usando un modelo de humanoide basado en un conjunto de segmentos (cuerpos  
95 rígidos), unidos entre sí por articulaciones, cuyo desplazamiento depende de torques aplicados a  
96 dichos cuerpos, y sus parámetros se ajustan a partir de la evolución del algoritmo.

97 El presente informe, describe y analiza pormenorizadamente: en la sección 2, las herramientas  
98 aplicadas; en la 3, el modelo del humanoide utilizado; en la 4 y 5, los diferentes tipos de actuadores  
99 y funciones de partida y contorno; en la 6, el algoritmo genético; y en la 7 y 8, los resultados  
100 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 sistema de control es un dispositivo (o conjunto de) que maneja, dirige o regula el comportamiento de otros dispositivos o sistemas, para minimizar los fallos y obtener los resultados deseados.

<sup>3</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>4</sup>Los algoritmos genéticos son métodos adaptativos, que pueden ser utilizados para resolver problemas de búsqueda y optimización. Están basados en la teoría de la selección natural, planteada por Charles Robert Darwin en 1859.

## <sup>101</sup> 2. Herramientas

### <sup>102</sup> 2.1. Motor físico

<sup>103</sup> Se llama motor físico o *physics engine* a un “*software* capaz de realizar simulaciones de  
<sup>104</sup> ciertos sistemas físicos, como la dinámica del cuerpo rígido, el movimiento de un fluido y la  
<sup>105</sup> elasticidad”[4].

<sup>106</sup> Actualmente, existen muchos motores físicos: ya sea de código propietario (*PhysX*, *Havok*), como  
<sup>107</sup> *open-source* (*Bullet Physics*, *Box2D*, *Newton*, *OGRE*). Considerando análisis relacionados [5][6],  
<sup>108</sup> y el hecho de que el espacio simulado fuese en 3D, se decidió que *Bullet Physics*[7] es el más  
<sup>109</sup> idóneo. Está implementado en C++ y ha sido utilizado en varios juegos (*Grand Theft Auto IV*  
<sup>110</sup> y *V*, etc); en los efectos especiales de películas (*Hancock*, *Bolt*, etc.); y proyectos científicos, como  
<sup>111</sup> la herramienta *open-source Tensegrity Robotics Toolkit* de la NASA<sup>5</sup>; entre otros.

<sup>112</sup> Si bien (como se verá más adelante) *Bullet* tiene problemas asociados con el coeficiente de restitu-  
<sup>113</sup> ción, posee una muy buena *performance* en la detección de colisiones, la dinámica y la resolución  
<sup>114</sup> de *constraints*. Esto se debe, en parte, a diferentes algoritmos iterativos de orden lineal (donde  
<sup>115</sup> el más importante es *Sequential Impulse*), de *caching* y también a la utilización de un modelo  
<sup>116</sup> de fricción de Coulomb aproximado [8]. Además, el motor físico brinda la posibilidad de regular  
<sup>117</sup> la precisión requerida en estos cálculos (sin olvidar que, con iguales recursos, a mayor preci-  
<sup>118</sup> sión, mayor capacidad de cómputo requerida y, ergo, mayor tiempo). Dado que la construcción  
<sup>119</sup> del humanoide implica definir características y restricciones de movimiento de cada uno de sus  
<sup>120</sup> segmentos, lo antes mencionado fue crucial para la elección de *Bullet Physics* en este proyecto.

#### <sup>121</sup> 2.1.1. Funcionamiento

<sup>122</sup> El motor físico se encarga de la simulación de cuerpos rígidos y la interacción entre los mismos.  
<sup>123</sup> En particular debe calcular el resultado de colisiones, arreglar el solapamiento de los cuerpos en  
<sup>124</sup> el espacio de simulación, estimar las fuerzas producidas debido a la fricción, y mantener el  
<sup>125</sup> cumplimiento de restricciones que puedan existir entre los cuerpos (por ejemplo, un vínculo para  
<sup>126</sup> formar una articulación). Para lograrlo, *Bullet* modela, a partir de un conjunto de ecuaciones,  
<sup>127</sup> las distintas restricciones que deben ser respetadas.

<sup>128</sup> Estos modelos reciben como dato la velocidad lineal y angular de cada objeto, y las fuerzas  
<sup>129</sup> que actúan sobre los mismos. Dada esta información, se resuelve el sistema de ecuaciones, cuya  
<sup>130</sup> solución representa las magnitudes de las fuerzas a accionar sobre el mismo, a fin de satisfacer  
<sup>131</sup> todas las restricciones. Para encontrar esta solución, entran en juego los distintos métodos de  
<sup>132</sup> complejidad lineal mencionados en la subsección anterior.

<sup>133</sup> Este procedimiento, se lleva a cabo en cada *timestep* de la simulación; donde un *timestep* es el  
<sup>134</sup> intervalo de tiempo que transcurre entre un cálculo de magnitudes y otro. Mientras menor sea  
<sup>135</sup> el *timestep*, el simulador será más preciso pero también consumirá más tiempo de cómputo.

#### <sup>136</sup> 2.1.2. Modelo de fricción y su verificación

<sup>137</sup> Hay reglas físicas relacionadas con el entorno y que son muy importantes para la caminata:  
<sup>138</sup> el modelo de fricción, con sus respectivos coeficientes de fricción y restitución.  
<sup>139</sup> En base a los modelos físico-matemáticos que representan a cada uno de los dos fenómenos en  
<sup>140</sup> cuestión (y que se explicarán a continuación), y pensando en posibles futuras simulaciones de

<sup>5</sup><http://bulletphysics.org/Bullet/phpBB3/viewtopic.php?f=17&t=9978>

141 varios humanoides chocando e interactuando entre sí; se llevaron a cabo dos ensayos para verificar  
 142 el funcionamiento del simulador físico *Bullet*.

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

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

148

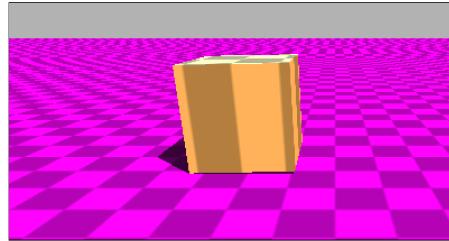


Figura 1: Visualización del sistema del cubo

149 Para esta prueba se utilizó el modelo matemático que representa la posición del cuerpo en el eje  
 150 horizontal en función del tiempo  $t$ , representado por la siguiente ecuación:

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

151 En este caso, el cuerpo empieza su movimiento en el origen, por lo tanto la posición inicial ( $x_i$ )  
 152 es cero.  $v_i$  es la velocidad inicial, y  $a$ , la aceleración. Debido a la fricción entre el cuerpo y el  
 153 suelo, se genera una fuerza de rozamiento  $\mathbf{F}_{\mu_d}$ <sup>6</sup> (ec. (2)) en la misma dirección que la velocidad  
 154 del sólido y en sentido contrario.

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

155 donde  $\mathbf{F}_N = mg$  es la fuerza normal que actúa sobre la caja de masa  $m$  por acción de la gravedad  
 156  $g = 10 \frac{m}{s^2}$ , y  $\mu_d$  es el coeficiente de fricción dinámico.

157

158 Finalmente, se obtiene la aceleración:

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

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

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

161

162 El intervalo de tiempo físico (o *timestep*) utilizado es  $\Delta t = \frac{1}{60} s$ . El *timestep* de animación (es  
 163 decir, cada cuánto tiempo se guardan en un archivo los datos logrados) es  $\Delta t' = 0.1$  y el tiempo  
 164 de simulación es de  $s = 100\Delta t$ .

165 Seguidamente, se muestran los gráficos obtenidos al correr los experimentos numéricos con los  
 166 siguientes valores:  $v_i = \{1, 3 \text{ y } 10\} \frac{m}{s}$  y  $\mu_d = \{0.25, 0.50 \text{ y } 0.75\}$ . Se compara la distancia en el  
 167 eje Z de la caja en Bullet, con su distancia en el mismo eje según la ec. (4).

<sup>6</sup>En las ecuaciones, los vectores se escriben en negrita.



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)  $\mu_d = 0.75$

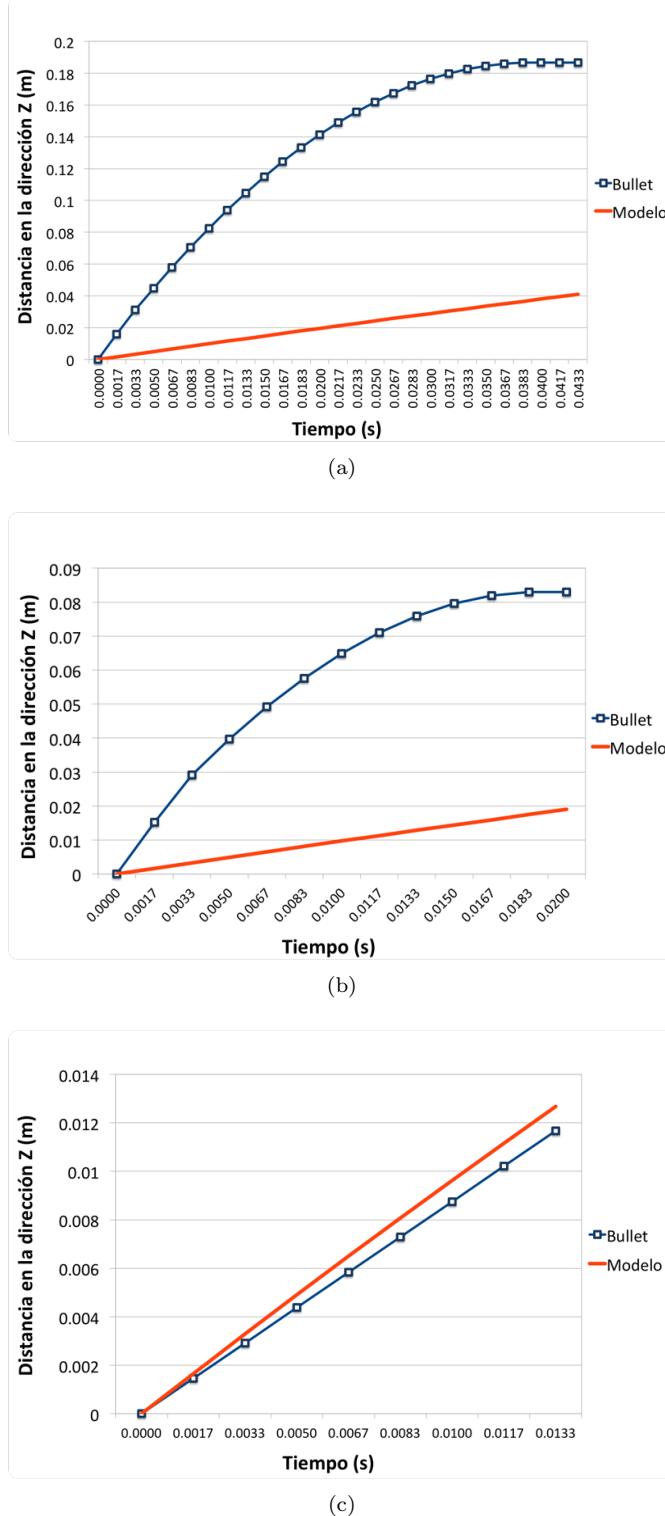


Figura 4: Resultados logrados de simular el sistema descripto en Fig. 1, usando  $v_i = 1 \frac{m}{s}$  y:

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

<sup>168</sup> Las Fig. 2 y 3 demuestran que los resultados de las pruebas fueron favorables. Los valores  
<sup>169</sup> obtenidos a partir de la simulación se corresponden (en mayor o menor medida) con los alcanzados  
<sup>170</sup> a partir del modelo matemático. Esto es un indicador de que Bullet debe estar usando dichos

modelos para ejecutar las simulaciones. Vale observar que, cuanto mayor es la velocidad inicial ( $v_i$ ), mayor es la similitud entre los dos casos.

No obstante, los gráficos que corresponden a la Fig. 4, presentan una discrepancia mayor entre la simulación y el modelo. Este hecho puede deberse al mecanismo utilizado por *Bullet* para resolver la fricción de un cuerpo a velocidades muy bajas.

### 2.1.2.2. Verificación del coeficiente de restitución

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

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

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

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

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

192

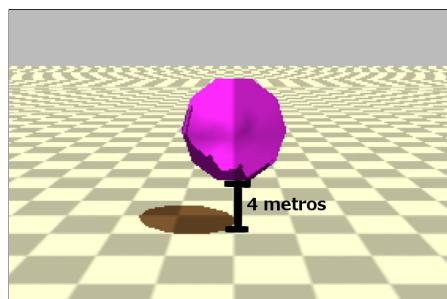


Figura 5: Visualización del sistema de la esfera

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

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

198

En la Tabla 1 se puede observar lo arrojado por los experimentos numéricos que se efectuaron; usando:  $v_i = \{-0.5, -3.5, -4, -5\} \frac{\text{m}}{\text{s}}$  y  $e_{sim} = \{0.2, 0.5, 0.8\}$ .

$v_i$	$e_{sim}$	$v_f$	$e_{medida}$	$\epsilon_{rel}$
$-0.5 \frac{m}{s}$	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 ( $e_{medida}$ ) de simular el sistema descripto en Fig. 5

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

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

### 212 2.1.3. Ventajas

213 Las ventajas del motor físico son:

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

### 217 2.1.4. Desventajas

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

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

- 222 – Utilizar una librería gráfica como *OpenGL* acoplada a una simulación de *Bullet*, puede pro-  
223 ducir resultados distintos, que si se usa un programa de visualización externo como OVITO.

## 224 **2.2. Librería de algoritmos genéticos**

225 Se utilizó la conocida librería de algoritmos genéticos para C++ GaLib, desarrollada por  
226 Matthew Wall del MIT [9].

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

229 Cabe aclarar que, antes de optar por GaLib, se había adoptado la librería Kataklinger, pero  
230 finalmente fue descartada, debido a errores o *bugs* en la misma (y que, cada vez, resultaban ser  
231 más inmanejables).

## 232 **2.3. Código fuente**

233 Al estar *Bullet* implementado en C++, el código fuente también está desarrollado en ese  
234 lenguaje. En *Bullet*, se define un *World* (o mundo físico) en donde se puede insertar, entre otras  
235 cosas, cuerpos rígidos. En este caso en particular, el mundo consta de un plano (el suelo), y el  
236 bípedo (compuesto por cuerpos rígidos y otros elementos físicos) ubicado sobre él.

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

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

## 243 **3. Modelo utilizado**

244 Dentro de los diversos modelos existentes ([1][2]), en este trabajo se procuró utilizar uno que  
245 fuera sencillo pero representativo a la vez.

246 Se modela al cuerpo del bípedo, con el motor *Bullet Physics*, como un conjunto de segmentos  
247 unidos entre sí por articulaciones. A cada uno de ellos se les aplica un torque en el centro de  
248 masa de cada segmento (denominado actuador). Que la caminata se produzca o no, depende  
249 del tipo de actuador empleado (la función utilizada para el torque), y de sus parámetros. El  
250 objetivo es encontrar dichos parámetros. Para eso se usan los algoritmos genéticos, un método  
251 de inteligencia artificial. De este modo se obtiene, de forma análoga a la selección natural, los  
252 individuos que mejor se adapten a la caminata. Tanto los actuadores como el algoritmo genético  
253 se explicarán más adelante.

### 254 **3.1. Composición física del humanoide**

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

257 Sobre la base de la anatomía humana, se dividió el cuerpo del humano virtual en: cabeza, tronco,  
258 miembro superior (brazo, antebrazo y mano), pelvis y miembro inferior (muslo, pierna y pie).

259 A los fines de este proyecto, sólo se consideraron pelvis y miembro inferior (Fig. 6).

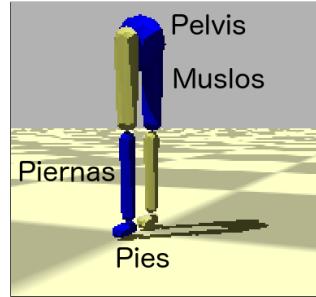


Figura 6: Segmentos del humanoide

260 En la Tabla 2 se presenta la composición física de cada uno de los segmentos del humano virtual,  
261 de acuerdo a la biomecánica [10].

Segmento		Cantidad	Forma	Largo (en m)	Masa (en kg)	Uniones
Miembro inferior	Pelvis	1	Esférica	0.08655	9.9718	Cadera
	Muslo	2	Esfero-cilíndrica	0.4015	10.3368	Cadera y Rodilla
	Pierna	2	Esfero-cilíndrica	0.4015	3.1609	Rodilla y Tobillo
	Pie	2	Esfero-cilíndrica	-	1.0001	Tobillo

Tabla 2: Composición física de cada segmento del humanoide

### 3.2. Articulaciones

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

263 A su vez, para cada caso en particular, se definieron cotas para los ángulos que pueden existir entre los segmentos (ver Tabla 3). Esto es muy importante, no sólo porque se adecúa a datos biológicos, sino porque, de otro modo la caminata no podría lograrse: si los ángulos son demasiado altos, la caminata se produce girando las piernas por encima de la pelvis; si por el contrario, son demasiado bajos, las piernas van a estar muy rígidas, originando pocos pasos y muy cortos.  
264 Asimismo, se le impide rotar a la pelvis, y se restringe la amplitud con la que puede moverse la cadera (de  $-\frac{\pi}{4}$  a  $\frac{\pi}{4}$ ). Esto se realiza porque, en caso contrario, el bípedo necesitaría un sistema de control para mantener el equilibrio, y eso excede el alcance de este trabajo.

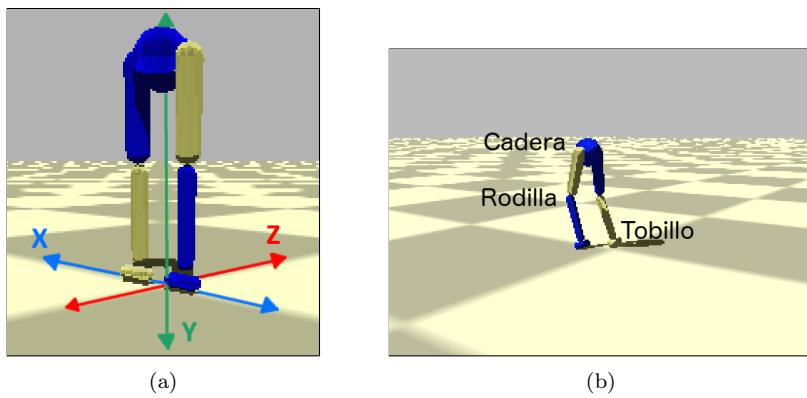


Figura 7: Humanoide diseñado: (a) ejes, y (b) articulaciones

		Ángulo mínimo	Ángulo máximo
Articulación	Cadera	$-\frac{\pi}{4}$	$\frac{\pi}{4}$
	Rodilla	$-\pi$	$\pi$
	Tobillo	0	0

Tabla 3: Rango de valores de ángulo de cada articulación del humanoide

## 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.

Es necesario aclarar que la caminata producida por el humanoide es plana (en 2D). Esto se debe a que la trayectoria pensada para el bípedo es una línea recta, y logrando que los segmentos se muevan en un solo eje es suficiente para cumplir con dicha trayectoria. También contribuye el hecho de que el torque se aplique en una sola dimensión. Por otra parte, los actuadores definidos a continuación son periódicos, y por eso no se pueden aplicar en el eje Z de los segmentos (se necesitarían actuadores reactivos, para poder detectar cuando el humanoide se está cayendo, etc.). Para indicar el módulo de dicho torque, se diseñaron diferentes funciones (todas ellas periódicas), mencionadas en las subsecciones que siguen.

288

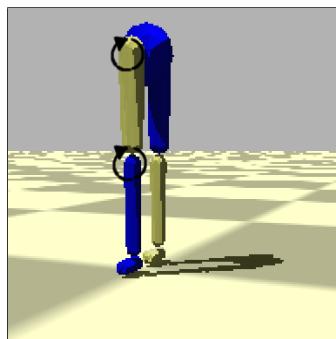


Figura 8: Aplicación de los actuadores en los segmentos del bípedo

### 4.1. Genérico

Es el actuador más sencillo, tanto matemática como computacionalmente.

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

donde  $f(t)$  es la función del actuador evaluada en el instante de tiempo  $t$ ,  $A_1$  y  $A_2$  son amplitudes,  $\omega_1$  y  $\omega_2$  son frecuencias (en  $\frac{1}{s}$ ),  $\phi$  es la fase en radianes, y  $C$  es un término independiente (ver Fig. 9 (a) y (b)).

La fase  $\phi$  es la misma en el seno y en el coseno, para evitar que se formen otro tipo de funciones no cíclicas.

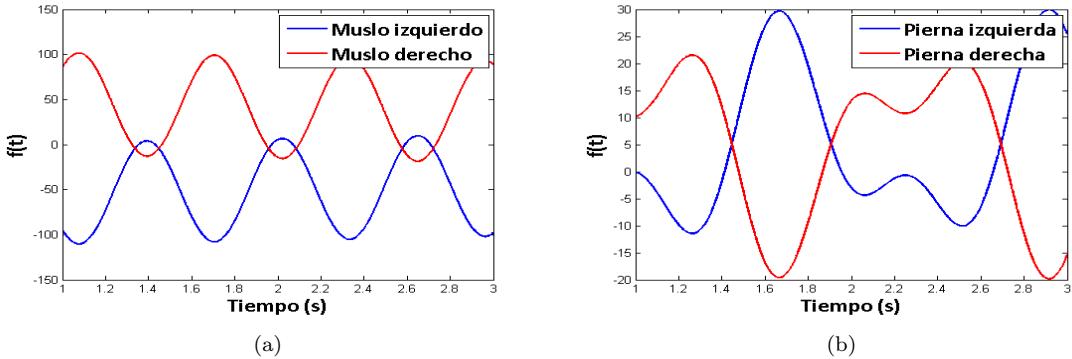


Figura 9: Ejemplo de actuador genérico aplicado en: (a) muslo, y (b) pierna

## 296 4.2. Coseno doble frecuencia

Basada en [3], esta función periódica utiliza medio ciclo de una función sinusoidal, y medio ciclo de otra (ambas pueden tener frecuencias distintas). Esto podría tener sentido porque en una caminata, un miembro inferior primero avanza hacia adelante y luego se extiende hacia atrás, y es razonable que esos dos movimientos se produzcan a frecuencias distintas (Fig. 10 (a) y (b)). La idea es lograr una función periódica a partir de una que no lo es (ya que  $t$  es lineal). Para eso, se utiliza la función  $\psi(t)$  (ec. (8)) que aplica una transformación a los números reales, para que se encuentren dentro del rango del ciclo completo (con las dos frecuencias).  $\omega$  es la frecuencia de  $f(t)$  (ec. (10)), que utiliza medio ciclo con frecuencia  $\omega_1$  y medio ciclo con frecuencia  $\omega_2$ .

$$\psi(t) = t + \phi - \left\lfloor \frac{t + \phi}{\pi/\omega_1 + \pi/\omega_2} \right\rfloor (\pi/\omega_1 + \pi/\omega_2) \quad \psi : \mathbb{R} \rightarrow \left[0, \frac{2\pi}{\omega}\right] \quad (8)$$

$$\omega = \frac{2\omega_1\omega_2}{\omega_1 + \omega_2} \quad (9)$$

$$f(t) = \begin{cases} A \cos(\omega_1 \psi(t)) + C & \text{si } \omega_1 \psi(t) < \pi \\ A \cos(\omega_2(\psi(t) - (\pi/\omega_1) + (\pi/\omega_2))) + C & \text{en otro caso} \end{cases} \quad (10)$$

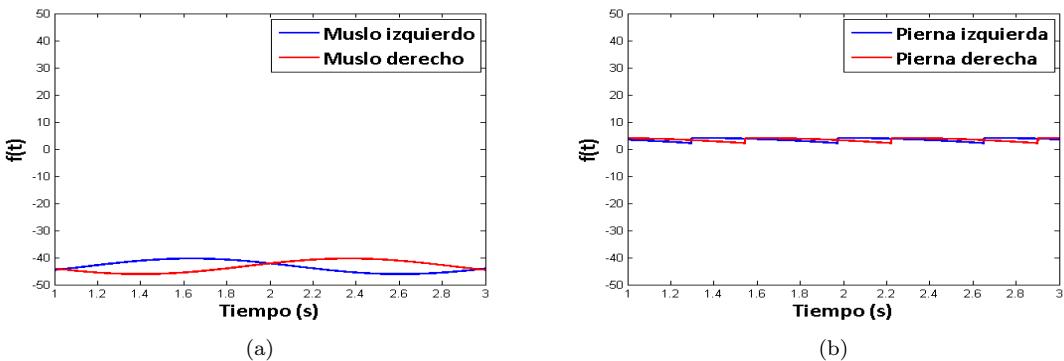


Figura 10: Ejemplo de actuador coseno doble frecuencia aplicado en: (a) muslo, y (b) pierna

### 309 4.3. Fourier de orden 2

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

$$f(t) = A_1 \sin(\omega t + \phi) + B_1 \cos(\omega t + \phi) + A_2 \sin(2\omega t + \phi) + B_2 \cos(2\omega t + \phi) + C \quad (11)$$

311 donde  $A_1, A_2, B_1$  y  $B_2$  son amplitudes y  $\omega$  es frecuencia (en  $\frac{1}{s}$ ) (ver Fig. 11 (a) y (b)).

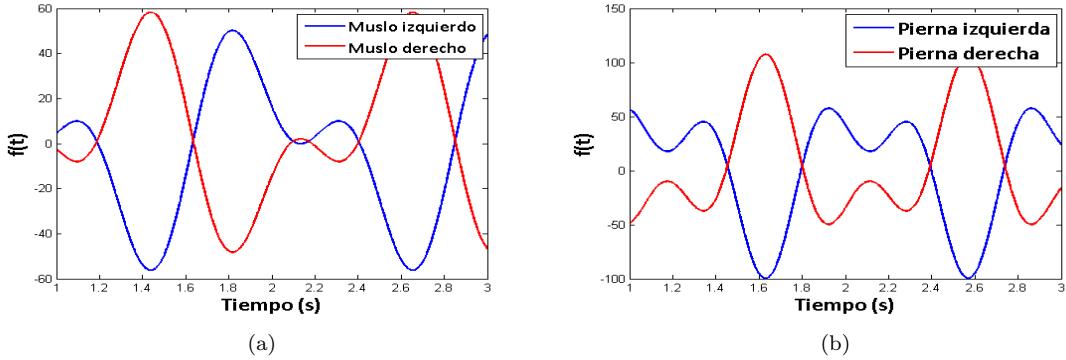


Figura 11: Ejemplo de actuador fourier de orden 2 aplicado en: (a) muslo, y (b) pierna

### 312 4.4. Fourier de orden 9

313 Es una extensión del actuador anterior, pero con 9 términos. Por ser de mayor grado, brinda  
314 una mayor precisión. Sin embargo, es más difícil de manejar computacionalmente; y, además, que  
315 sea más preciso no garantiza que con él se pueda lograr una buena caminata (ver Fig. 12 (a) y (b)).

$$\begin{aligned} f(t) = & A_1 \sin(\omega t + \phi) + B_1 \cos(\omega t + \phi) + A_2 \sin(2\omega t + \phi) + B_2 \cos(2\omega t + \phi) \\ & + A_3 \sin(3\omega t + \phi) + B_3 \cos(3\omega t + \phi) + A_4 \sin(4\omega t + \phi) + B_4 \cos(4\omega t + \phi) \\ & + A_5 \sin(5\omega t + \phi) + B_5 \cos(5\omega t + \phi) + A_6 \sin(6\omega t + \phi) + B_6 \cos(6\omega t + \phi) \\ & + A_7 \sin(7\omega t + \phi) + B_7 \cos(7\omega t + \phi) + A_8 \sin(8\omega t + \phi) + B_8 \cos(8\omega t + \phi) \\ & + A_9 \sin(9\omega t + \phi) + B_9 \cos(9\omega t + \phi) + C \end{aligned} \quad (12)$$

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

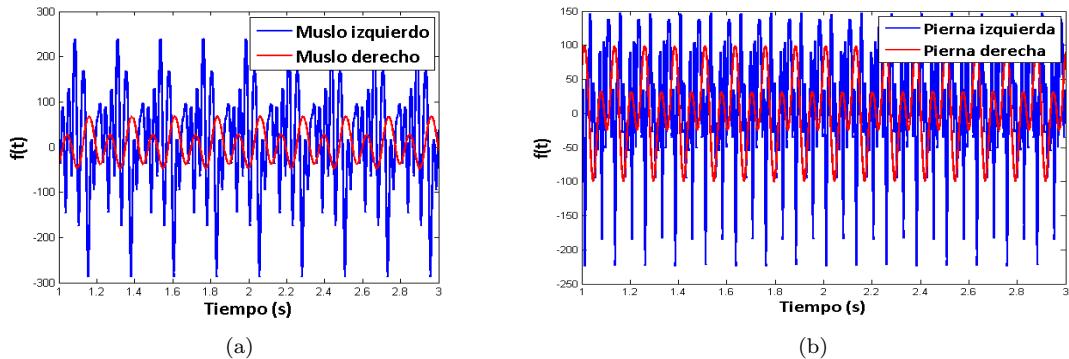


Figura 12: Ejemplo de actuador fourier de orden 9 aplicado en: (a) muslo, y (b) pierna

## 318 5. Condiciones iniciales y de contorno

319 Como las funciones periódicas señaladas en los actuadores, no fueron suficientes para lograr  
 320 la caminata, se le adosaron las funciones que se detallan seguidamente.

### 321 5.1. Función partida

322 El andar del humanoide es cíclico. Sin embargo, por la posición inicial del individuo, se requiere  
 323 para el tiempo del primer paso, una función distinta a la del resto de la caminata. El tipo de  
 324 función puede ser cualquiera de los actuadores vistos anteriormente (pero no necesariamente con  
 325 los mismos valores de amplitud, frecuencia y fase asignados a las piernas). Empero, se utilizó la  
 326 función vista en el actuador genérico.  
 327 Por otra parte, para simplificar el modelo, se decidió que el tiempo considerado para el primer  
 328 paso sea fijo, y de 0.7 segundos. Dicho valor fue extraído de forma experimental.

### 329 5.2. Fase sincronizada

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

$$333 \quad f_i(t) = f(t) \quad (13)$$

$$334 \quad f_d(t) = f(t + \frac{\pi}{2}) \quad (14)$$

335 siendo  $f(t)$  la función de movimiento (o actuador) en el momento  $t$ , y  $f_i$  y  $f_d$  las funciones de la  
 336 pierna izquierda y derecha, respectivamente.  
 337

## 338 6. Algoritmo genético

339 Los algoritmos genéticos son métodos adaptativos, que pueden ser utilizados para resolver  
 340 problemas de búsqueda y optimización. Están inspirados en la teoría darwiniana de la selección  
 341 natural. La entidad a optimizar constituye un individuo dentro de una población; que puede  
 342 ser cruzado con otros individuos para obtener un “hijo”, que presenta características de sus  
 343 “padres”. A través de una función de *fitness* se decide cuáles de ellos son los más “aptos”, que  
 344 luego pasarán a una nueva generación de individuos. Este proceso se repite hasta obtener un  
 345 individuo lo suficientemente apto, que representa a la entidad optimizada.

346 El objetivo de este proyecto es optimizar la caminata del humanoide. Ergo, en el algoritmo  
 347 genético aplicado, el individuo está definido por rasgos del bípedo, y la función de *fitness* se  
 348 focaliza en mejorar diversos aspectos de dicha caminata.

### 349 6.1. Individuo

350 La información genética de cada individuo, está definida a partir de un vector que contiene de  
 351 forma contigua, los parámetros de la función partida (optativa) y los asociados a los actuadores  
 352 (obligatorios), que actúan sobre cada uno de los segmentos (muslos y piernas) del humano virtual.

353 **6.1.1. Parámetros**

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

357 La Fig. 13 muestra la estructura del individuo y su composición para un humanoide con función  
 358 partida y actuador genérico.

359

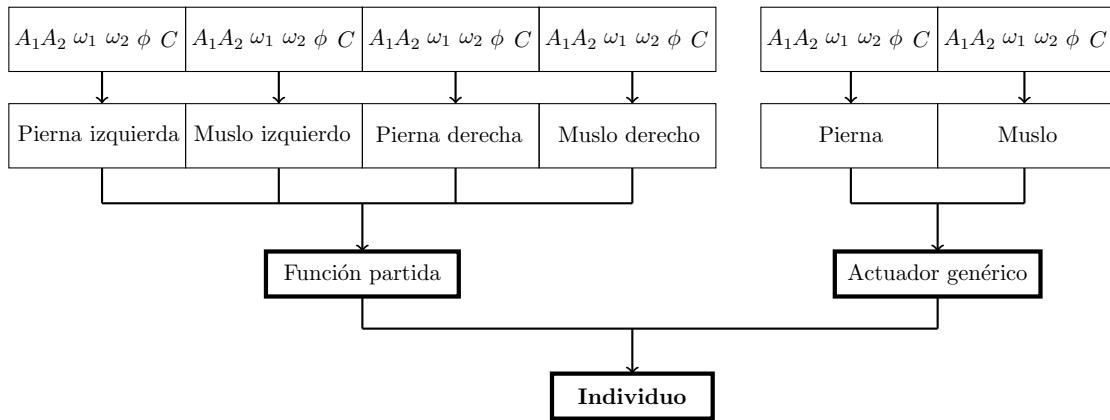


Figura 13: Esquema de un individuo - Ejemplo

360 **6.1.2. Valores**

361 Cada uno de los segmentos tiene una composición física distinta (largo, masa, etc.), razón por  
 362 la cual no necesariamente sus genes deban tener los mismos rangos de valores, tal como puede  
 363 apreciarse en las Tablas 5 y 4.

364

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 4: 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
Coseno doble frecuencia	Muslo	Amplitud	-50	50
	Pierna	Amplitud	-30	30
	Muslo y Pierna	Frecuencia	0.01	5
	Muslo y Pierna	Fase	$-\pi$	$\pi$
	Muslo y Pierna	Término independiente	-30	30
Fourier de orden 2	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
Fourier de orden 9	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

Tabla 5: Rango de valores que puede tomar cada gen, según el tipo de actuador

### 365 6.1.3. Implementación de individuos

366 Para favorecer el análisis de las distintas características arriba indicadas, se implementaron  
 367 varios individuos, cada uno de ellos con propiedades distintas (Tabla 6).

368

	Individuo				
	Tipo 1	Tipo 2	Tipo 3	Tipo 4	Tipo 5
Actuador	Genérico	Genérico	Coseno doble frecuencia	Fourier de orden 2	Fourier de orden 9
Función partida	No	Sí	Sí	Sí	Sí
Fase sincronizada	Sí	Sí	Sí	Sí	Sí

Tabla 6: Tipo de individuos

### 369 6.1.4. Constitución del cromosoma

370 Los distintos actuadores y la función partida tienen los parámetros presentados en la sección  
 371 6.1.1. Sus respectivas cantidades pueden verse en la Tabla 7.

372 En la Tabla 8 se muestra la composición del cromosoma de cada individuo, que depende de los  
 373 actuadores y la función partida usados. En ella se puede observar cómo según el tipo de individuo,  
 374 varía la cantidad de genes, es decir, la longitud del cromosoma. Vale aclarar que la función  
 375 partida se especifica para cada segmento (los dos muslos y las dos piernas); en cambio, para los

<sup>376</sup> actuadores, sólo se definen dos (uno para los muslos y otro para las piernas).

<sup>377</sup>

		Parámetro			
		Amplitud	Frecuencia	Fase	Término independiente
Actuador	Genérico	2	2	1	1
	Coseno doble frecuencia	1	2	1	1
	Fourier de orden 2	4	1	1	1
	Fourier de orden 9	18	1	1	1
	Función partida	2	2	1	1

Tabla 7: Cantidad de parámetros según tipo de actuador y función partida

Parámetro	Individuo				
	Tipo 1	Tipo 2	Tipo 3	Tipo 4	Tipo 5
Amplitud	4	12	8	16	44
Frecuencia	2	6	8	6	6
Fase	2	6	6	6	6
Término independiente	2	6	6	6	6
Totales	10	30	28	34	62

Tabla 8: Cantidad de parámetros según tipo de individuos

## 6.2. Fitness

<sup>379</sup> El papel de la función de *fitness* ( $F$ ) en un algoritmo genético es evaluar qué tan bueno es un  
<sup>380</sup> individuo. En este caso, está definida como un producto de cinco módulos o propiedades: altura  
<sup>381</sup> ( $H$ ), velocidad ( $V$ ), dirección ( $D$ ), simetría ( $S$ ) y pies abajo ( $PA$ ):

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

<sup>382</sup> Los cinco tienen la misma importancia y por eso, como se verá a continuación, están definidos de  
<sup>383</sup> forma similar (con una función exponencial y pueden valer entre 0 y 1). Con todo esto, dado que  
<sup>384</sup> el *fitness* está pensado como un producto, basta con que uno de los módulos sea muy chico para  
<sup>385</sup> “anular” al individuo (es decir, otorgarle un valor que tiende a cero). Sin embargo, los diferentes  
<sup>386</sup> módulos no son completamente independientes entre sí: por ejemplo, si la altura es demasiado  
<sup>387</sup> baja, posiblemente la velocidad y la dirección no sean adecuadas.

### 6.2.1. Altura

<sup>388</sup> Es un factor relacionado con la altura del individuo en toda la simulación, y se expresa:

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

<sup>391</sup>  
<sup>392</sup> 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  
<sup>393</sup> tiempo  $t_n$ ,  $N$  la cantidad pasos de simulación y  $C$  una constante  $C = 5$ .

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

### 399 6.2.2. Velocidad

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

$$402 V = \frac{\sum_{n=0}^T e^{-C(v_{t_n} - V_O)^2}}{N} \quad (17)$$

403 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  
 404 tiempo  $t_n$ , y  $V_O$  la velocidad objetivo en el eje Z (el eje de la caminata).  
 405 Sigue una lógica y cálculo similares al factor de altura: a mayor discrepancia de la velocidad  
 406 real del humanoide con  $V_O$ , menor (y más cercano a cero) es el valor arrojado por el módulo de  
 407 velocidad.  
 408

### 409 6.2.3. Dirección

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

$$412 D = \frac{\sum_{n=0}^T e^{-C(\mathbf{v}_{t_n} \cdot \mathbf{v}_O - 1)^2}}{N} \quad (18)$$

413 donde  $t_0$  es el tiempo inicial,  $t_T$  el tiempo final,  $\mathbf{v}_{t_n}$  el versor de la dirección del humanoide en  
 414 el momento  $t_n$  y  $\mathbf{v}_O$  el versor de la dirección objetivo.  
 415 El producto escalar entre los versores responde a la Similitud Coseno:  $\cos \theta = \frac{\mathbf{A} \cdot \mathbf{B}}{AB}$ , donde  $\mathbf{A}$  y  
 416  $\mathbf{B}$  son vectores que no se encuentran normalizados, y  $\theta$  es el ángulo formado entre ellos. Así, si  
 417  $\cos \theta = 1$ , significa que los vectores están paralelos entre sí (que es el efecto buscado en el caso  
 418 de la dirección).  
 419 Al producto escalar se le resta 1, para que el módulo sea consistente con la función exponencial  
 420 utilizada y que valga 1 cuando  $\theta = 0$ , y 0 cuando  $\theta = \pi$ . Cabe aclarar que se trata al ángulo en  
 421 forma simétrica, ya que, por ejemplo  $\cos(-\pi/6) = \cos(\pi/6)$ .

### 421 6.2.4. Simetría

422 Este indicador marca qué tan equidistantes se encuentran los pies de la cadera, a lo largo de la  
 423 caminata. Aplicando solamente los módulos antes mencionados, provocaba resultados en donde  
 424 una pierna quedaba más distante de la pelvis que la otra, lo que producía que el humanoide se  
 425 terminara arrastrando, posiblemente afectando a la velocidad.  
 426 Para mayor simplicidad, la simetría  $S$  se calculó a partir de los pies (y no de las piernas). Se  
 427 tomaron en cuenta sólo los ejes X y Z, porque son los relacionados a la velocidad y a la dirección,  
 428 respectivamente.

$$429 S = \frac{\sum_{n=0}^T \frac{1}{2}[e^{-C(|lf_z + rf_z|^2)} + e^{-C(|lf_x + rf_x|^2)}]}{N} \quad (19)$$

429

430 donde  $lf_x$  y  $lf_z$  es la distancia desde el pie izquierdo hasta la pelvis en los ejes X y Z, respectivamente; y en donde  $rf_x$  y  $rf_z$  es lo mismo, pero para el pie derecho.

#### 432 6.2.5. Pies abajo

433 Con los módulos señalados anteriormente, se busca que el humanoide camine con una veloci-  
434 dad y dirección determinadas, que no se caiga y que mantenga simetría mientras ejecuta sus  
435 movimientos. Pero, todo esto daría, en el mejor de los casos, una caminata estilo “estrella”.

436 Sin embargo, una característica fundamental en una caminata normal es que las piernas (ergo,  
437 los pies también) no sobrepasen la cadera. Si bien ésta es una propiedad negativa (expresa lo que  
438 no debe tener una caminata), y se puede correr el riesgo de restringir demasiado, su ausencia da  
439 resultados peores.

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

440

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

445

### 446 6.3. Operadores del algoritmo

447 Permiten controlar en detalle el proceso de optimización. En particular, se busca un balance  
448 entre la diversidad de los individuos, el aumento del *fitness* a lo largo del algoritmo, y evitar la  
449 convergencia hacia una población sobre la cual no se puede seguir mejorando.

#### 450 6.3.1. Métodos de selección

451 De la vasta cantidad de métodos de selección que existen, se utilizaron: ***Elite*** (en donde  
452 se elige el individuo con mayor aptitud de la población); y ***Roulette*** (método estocástico, que  
453 selecciona un individuo de la población total al azar, con una probabilidad proporcional a su  
454 *fitness*).

#### 455 6.3.2. Métodos de crusa

456 El método de crusa (o *crossover*) utilizado es el siguiente: De dos individuos (los padres), se  
457 originan dos nuevos individuos (los hijos). Se toma cada uno de los genes de los padres y se elige,  
458 con una probabilidad uniforme, uno de ellos para un hijo y el otro para el otro hijo.

459 La probabilidad de que este proceso ocurra es de 0.9.

#### 460 6.3.3. Mutación

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

464 Dicha probabilidad de mutación es de 0.3.

465 **6.3.4. Otras características**

466 En el algoritmo genético se utilizan 1000 generaciones, de 50 individuos cada una. Además,  
467 la simulación de cada individuo (necesaria para calcular el *fitness*) es de 4 segundos.

468 **7. Resultados obtenidos**

469 En la Fig. 14 se representan los individuos definidos en el algoritmo genético con sus carac-  
470 terísticas.

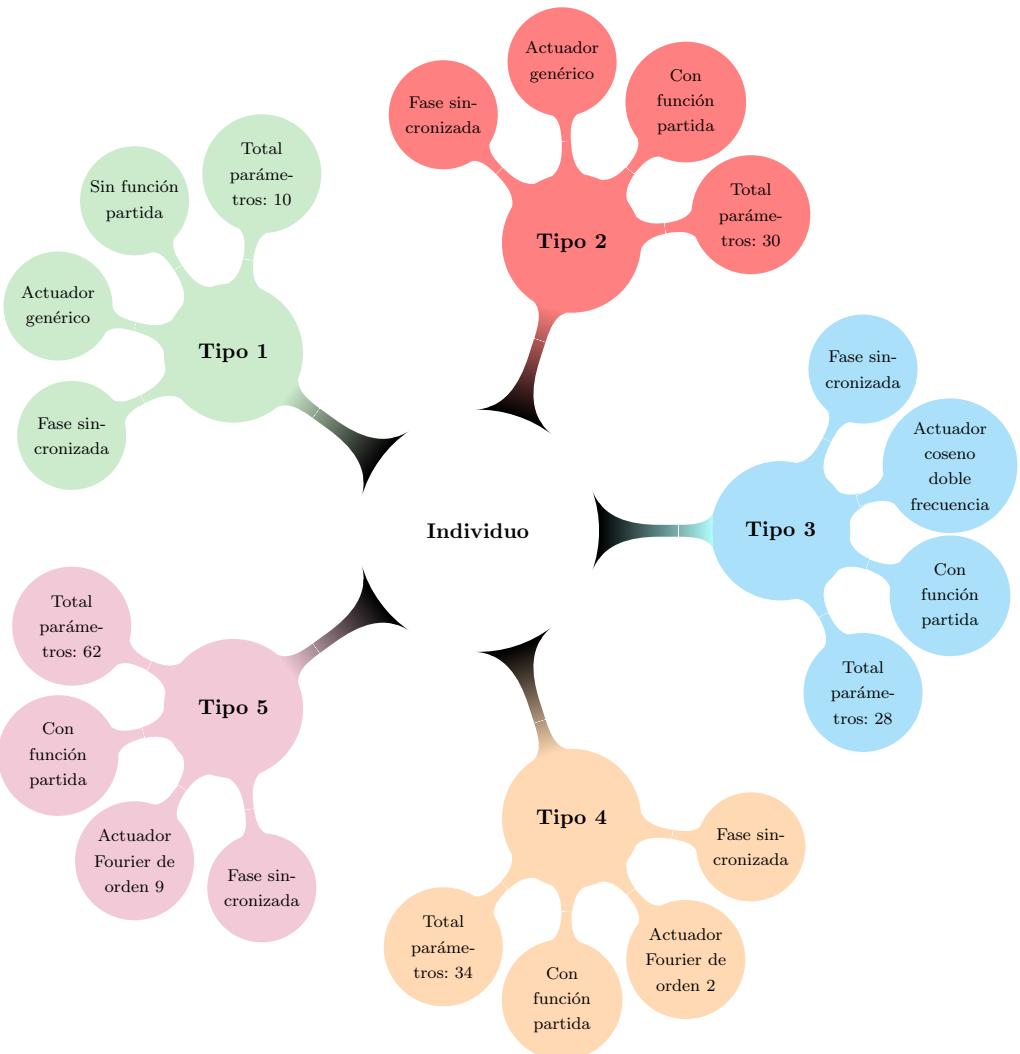


Figura 14: Individuos definidos en el algoritmo genético

471 Sobre ellos se realizaron pruebas, corriendo el algoritmo genético, y evaluando el resultado  
472 alcanzado posteriormente (ya sea numérica o gráficamente).  
473 A continuación se analizan distintos aspectos relevantes.

## 474 7.1. Evolución del *fitness* según tipo de individuos

475 Los individuos que usan actuadores de Fourier son los que producen *fitness* más bajos; el de  
 476 orden 2 (individuo de tipo 3) se estanca (al igual que el de tipo 1) a las pocas generaciones; el  
 477 *fitness* del individuo con actuadores Fourier de orden 9 se “ameseta” progresivamente (después  
 478 de 500 generaciones).

479 El individuo de tipo 1 es el individuo con *fitness* más alto, pero que se obtuvo a las pocas  
 480 generaciones (es decir que es un máximo local). El individuo de tipo 2 (con actuadores genéricos  
 481 y función partida) va aumentando su *fitness* progresivamente, aunque sin superar al de tipo 1.  
 482 Por otra parte, el individuo de tipo 5 (con actuadores coseno doble frecuencia), va mejorando su  
 483 *fitness* paulatinamente, lo que impide estancarse en un máximo local. Además, tiene el segundo  
 484 mejor *fitness*.

485 Puede notarse que los rangos manejados en la Fig. 15 casi alcanzan el 0.7 (y no a 1, su cota  
 486 superior). Esto se debe a que la función de *fitness* está definida como un producto de ciertos  
 487 módulos: suponiendo que cada uno de ellos estuviera al 92 %, se tendría  $0.92^5 = 0.659$ .

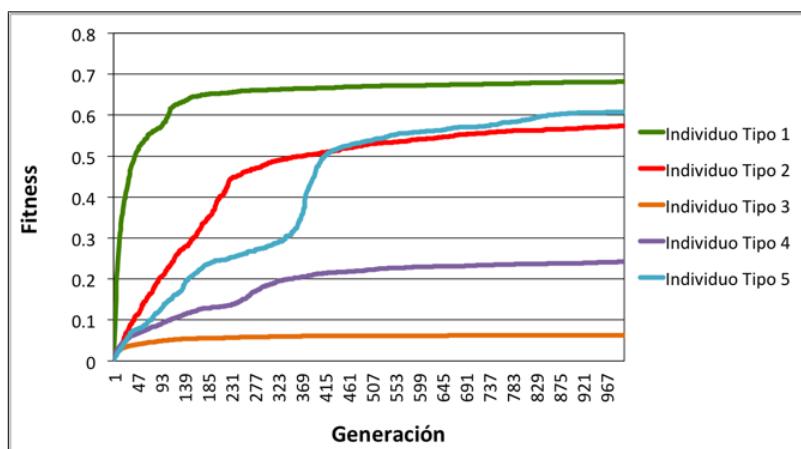


Figura 15: Evolución del *fitness* según tipo de individuos

## 488 7.2. Velocidad según tipo de individuos

489 Seguidamente, se muestra para cada individuo, su velocidad instantánea a lo largo del tiempo.

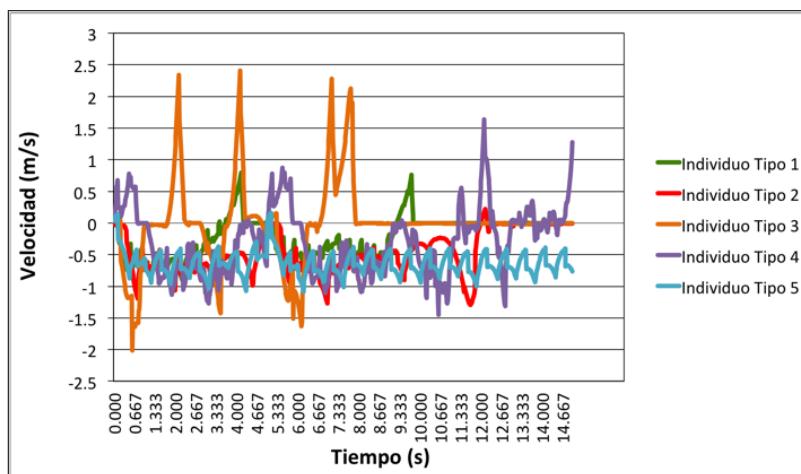


Figura 16: Velocidad según tipo de individuos

- 490 Como puede observarse en el caso de los individuos 3 y 4 (Fourier de orden 2 y 9, respectivamente),  
 491 se producen picos altos y pronunciados en la velocidad. Eso repercute en que la velocidad media no  
 492 sea  $1.3 \frac{m}{s}$  (que es la velocidad objetivo), y ergo, en el módulo de velocidad del *fitness* (provocando  
 493 que éste sea más bajo).
- 494 En el individuo de tipo 1, también se producen picos un poco menos pronunciados, pero luego  
 495 de los 10 segundos, no se registra ninguna velocidad. Eso sucede porque el humanoide queda  
 496 suspendido cuando intenta caer hacia atrás.
- 497 La velocidad del individuo de tipo 2 oscila de forma irregular, pero no tiene picos muy elevados.  
 498 Por último, en el individuo de tipo 5, la velocidad oscila de forma cíclica, continuada y armoniosa  
 499 (no hay picos altos).

### 500 7.3. Altura según tipo de individuos

501 Como puede identificarse en la Figura 17, la altura de los individuos es otra característica  
 502 para diferenciarlos en su rendimiento.

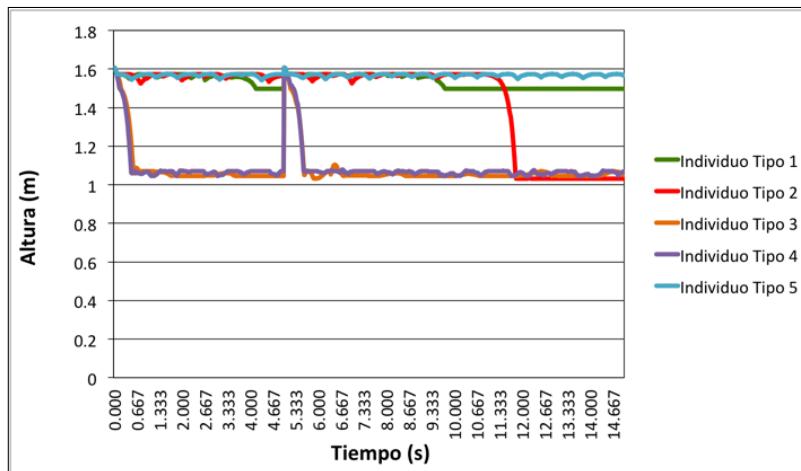


Figura 17: Altura (posición de la pelvis) según tipo de individuos

- 503 En efecto, los individuos de tipo 3 y 4 (que utilizan actuadores de Fourier de orden 2 y 9,  
 504 respectivamente), son los que caen de forma más abrupta, y aunque intentan levantarse, vuelven  
 505 a caer con la misma intensidad.
- 506 A su vez, el individuo de tipo 1 (actuador Genérico sin función partida) mantiene su altura,  
 507 hasta que a los 10 segundos queda suspendido a una altura levemente menor. Esto ocurre porque  
 508 se cae para atrás, y las limitaciones de la cadera impiden que caiga. El individuo de tipo 2, con  
 509 actuadores genéricos y función partida, logra mantener la altura, hasta que cae hacia adelante, en  
 510 cuclillas. Considerando que cuando cae no vuelve a levantarse (ya que en ese movimiento suelen  
 511 emplearse los brazos y manos), tiene un comportamiento similar a una caminata real.
- 512 En lo que respecta al individuo de tipo 5, que utiliza como actuadores la función coseno doble  
 513 frecuencia, mantiene su altura de forma constante, pero con oscilaciones leves y continuadas en  
 514 el tiempo. De los cinco, es el que da mayor cantidad de pasos.

### 515 7.4. Comparación de tipo de individuos

516 En base a lo visto anteriormente, queda claro que los actuadores de Fourier no dieron buenos  
 517 resultados tanto en fitness, como en altura y velocidad.

518 Por otra parte, es necesario incluir la función partida (es decir, separar a la caminata en un  
519 “primer paso”, y el ciclo), ya que si bien su ausencia puede originar *fitness* más altos, genera  
520 inestabilidad a los pocos segundos (como en el individuo de tipo 1).

521 El individuo de tipo 5 logra un movimiento cíclico y repetitivo, que deriva en una caminata *ad*  
522 *infinitum*, pero con el costo de que sea “robótica” (muy parecida a lo visto con un *passive walker*).

523 **7.5. Video**

524 Para una mejor visualización de los distintos individuos obtenidos, y de su evolución a lo  
525 largo de las generaciones del algoritmo genético, se acompaña video.

526 **Acá va el link del video**

527 **8. Conclusiones**

528 El objetivo principal de este proyecto fue lograr producir la simulación y animación bio-  
529 mecánica de la caminata de un humanoide virtual. Para eso, se eligió el motor físico *Bullet Physics*,  
530 lo que requirió no solo aprender sobre su funcionamiento y los métodos físicos implementados,  
531 sino también realizar pruebas para verificar qué tan próximos eran el modelo fisico-matemático  
532 ideado y el utilizado por Bullet.

533 Una vez modelado el humanoide, se implementaron individuos con diferentes características (en  
534 especial, actuadores), para facilitar la comparación. En la sección 7 se verifica que las funciones  
535 utilizadas en los actuadores son decisivas para lograr una caminata.

536 Los actuadores que mejores resultados dieron, fueron aquellos en donde se empleaba dos frecuen-  
537 cias  $\omega$  en vez de una.

538 Lo ocurrido con el individuo de tipo 1 (genérico y sin función partida), que es el que tiene un  
539 *fitness* más alto, aunque no produce una caminata acorde, posiblemente se deba a que el tiempo  
540 de simulación empleado en el algoritmo genético, fuera corto.

541 Se comprobó que los individuos con actuadores genérico y coseno doble frecuencia son los que  
542 mejor caminan, manteniendo su altura por más tiempo y con *fitness* más alto. Sin embargo, son  
543 caminatas muy distintas: la del primero resulta ser más natural, pero se cae más rápido; mientras  
544 que la del segundo es más “robótica” (ya que parece un *passive walker*), pero más estable (no se  
545 cae nunca ni se queda quieto).

546 La función de fitness que se plantea (con sus módulos) indica propiedades ideales para una cami-  
547 nata, pero que no implican realismo en la misma. Y entonces dos individuos con *fitness* parecidos  
548 dieron como fruto caminatas muy distintas.

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

551 Entre los trabajos a futuro para integrar a este proyecto, se encuentran lograr que la caminata  
552 se produzca en 3 dimensiones; y analizar el comportamiento de varios humanoides chocando e  
553 interactuando entre sí.

554 Se puede concluir que, la caminata de una persona, algo que parece simple y sencillo, muestra  
555 su verdadera complejidad cuando debe ser simulada por medio de actuadores aplicados a un  
556 conjunto de segmentos interconectados.

## Referencias

- 557 [1] Marek Wojtyra, *Multibody Simulation Model of Human Walking - Warsaw University of*  
559 *Technology, 2003*
- 560 [2] Thomas Geijtenbeek, Michiel van de Panne y A. Frank van der Stappen, *Flexible Muscle-*  
561 *Based Locomotion for Bipedal Creatures, 2013*
- 562 [3] Kevin Kenny, Máximo Videla y Axel Wassington, *Proyecto Final para la obtención del título*  
563 *Ingeniero en Informática: Simulación y animación de un cuadrúpedo virtual - ITBA, 2014*
- 564 [4] Wikipedia: [https://es.wikipedia.org/wiki/Physics\\_engine](https://es.wikipedia.org/wiki/Physics_engine)
- 565 [5] Andreas Gerndt y otros, *An Evaluation of Open Source Physics Engines for Use in Virtual*  
566 *Reality Assembly Simulations. Fecha de publicación: 2012*
- 567 [6] Tom Erez y otros, *Simulation Tools for Model-Based Robotics: Comparison of Bullet, Havok,*  
568 *MuJoCo, ODE and PhysX*
- 569 [7] Sitio web de Bullet Physics: <http://www.bulletphysics.org/>
- 570 [8] Erin Catto, *Iterative Dynamic with Temporal Coherence. Fecha de publicación: 2005*
- 571 [9] Sitio web de GaLib: <http://lancet.mit.edu/ga/>
- 572 [10] <http://www.exrx.net/Kinesiology/Segments.html>