

# Aportaciones y Aplicaciones de Disciplinas Bioinspiradas a la Creatividad Computacional



Carlos Alberto Sánchez Quintana

Dept. Lenguajes y Ciencias de la Computación

Universidad de Málaga, España

Tesis Doctoral

Para la obtención del grado

Doctor por la Universidad de Málaga

ABRIL, 2015



Publicaciones y  
Divulgación Científica

AUTOR: Carlos Alberto Sánchez Quintana  
EDITA: Publicaciones y Divulgación Científica. Universidad de Málaga



Esta obra está sujeta a una licencia Creative Commons:  
Reconocimiento - No comercial - SinObraDerivada (cc-by-nc-nd):  
[Http://creativecommons.org/licenses/by-nc-nd/3.0/es](http://creativecommons.org/licenses/by-nc-nd/3.0/es)  
Cualquier parte de esta obra se puede reproducir sin autorización  
pero con el reconocimiento y atribución de los autores.  
No se puede hacer uso comercial de la obra y no se puede alterar, transformar o hacer  
obras derivadas.

Esta Tesis Doctoral está depositada en el Repositorio Institucional de la Universidad de  
Málaga (RIUMA): [riuma.uma.es](http://riuma.uma.es)

**Dr. Francisco J. Vico**, Catedrático de Universidad, Departamento de Lenguajes y Ciencias de la Computación, Universidad de Málaga, España.

Certifica que:

**Carlos Alberto Sánchez Quintana**, Ingeniero Superior en Informática, ha completado su tesis doctoral:

**Aportaciones y Aplicaciones de  
Disciplinas Bioinspiradas a la  
Creatividad Computacional**

en el Departamento de Lenguajes y Ciencias de la Computación en la Universidad de Málaga, bajo su supervisión. He dirigido esta tesis doctoral, y la apruebo para ser presentada a la Comisión de Doctorado de la Universidad de Málaga.

Málaga, Abril 2015.

Dr. Francisco J. Vico.

Catedrático de Universidad

Universidad de Málaga

España



*A mis padres, por haberme dado la oportunidad de recorrer este camino maravilloso que es la vida.*

*Manuela, por dejarme recorrerlo junto a ella y ser mi fiel compañera.*

*Wendy, por su compañía y cariño incondicional hasta el final.*



## Resumen

¿Puede una computadora desarrollar un comportamiento creativo? Esta compleja cuestión ha despertado un creciente interés en las últimas décadas. Es un hecho evidente que las computadoras han superado la capacidad humana en múltiples dominios. Sin embargo, alcanzar la creatividad humana sigue suponiendo un reto para las computadoras. Por otra parte, la naturaleza ha sido una de las principales fuentes de inspiración para la propuesta de novedosas soluciones creativas en diferentes áreas y contextos. En este sentido, dentro de la inteligencia artificial, el paradigma de la computación evolutiva, inspirado en la evolución natural, ha propuesto varias técnicas, entre ellas los algoritmos evolutivos, que han sido empleadas ampliamente en procesos creativos. Por definición, la creatividad requiere amplias dosis de innovación y diversidad. En el campo de la biología, recientes hipótesis apuntan a que el proceso de desarrollo es un mecanismo fundamental en el surgimiento de innovación y diversidad en los seres vivos. Por este motivo, el campo de la biología evolutiva del desarrollo (*evo-devo*), ha emergido para reclamar su incorporación como componente clave en la evolución de una gran diversidad de comportamientos y diseños estructurales innovadores. En el campo de la computación, la biología evolutiva del desarrollo ha inspirado dos disciplinas: el desarrollo artificial, que incorpora el proceso de desarrollo en los algoritmos evolutivos mediante codificación indirecta del esquema genotipo-fenotipo; y la ingeniería embriomórfica, la cual, imitando el proceso de desarrollo biológico, persigue el desarrollo de morfologías y comportamientos complejos artificiales mediante la auto-organización de grandes cantidades de pequeños agentes.

En línea con la compleja cuestión planteada sobre la capacidad creativa de las computadoras y la inspiración de la naturaleza como fuente de creatividad e innovación, este trabajo de tesis explora la aplicación de diferentes disciplinas bioinspiradas, concretamente los algoritmos evolutivos, el desarrollo artificial y la ingeniería embriomórfica, para la generación de productos creativos con diferentes campos de aplicación. Para ello, se presentan modelos computacionales, cuyas soluciones son aplicables en contextos tan diversos como la composición algorítmica, la medicina, la animación por computador y la robótica.

Concretamente, en el campo de la composición algorítmica, se presenta *Melomics*, un nuevo enfoque en creatividad computacional basado en *evo-devo*, cuyo objetivo es la automatización completa del proceso de composición musical. Parte del contenido que aparece en esta tesis sobre este innovador enfoque ha sido publicado en [152]. Con el objetivo de medir en qué grado la música generada por la tecnología *Melomics* tiene la misma valencia que la compuesta e interpretada por un humano, se exponen los resultados de un estudio, los cuales determinan que la música compuesta por *Melomics* es en la práctica indistinguible de la música compuesta por un humano, tanto desde el punto de vista emocional como evocador, siendo los participantes incapaces de distinguir las piezas compuestas por humano y las piezas compuestas por *Melomics*. En el campo de la medicina, la aplicación de  $M^3$  (un sistema basado en *Melomics*) a terapia es estudiada mediante un experimento controlado orientado a la reducción del dolor en un entorno clínico, cuyos resultados han sido publicados en [142]. Por otro lado, en el campo de la animación por computador, se presenta un modelo evolutivo de avatares animados mediante física como soporte creativo para la sincronización automática de personajes y música en escenas de animación. Para su aplicación práctica, se propone una herramienta para la importación de los resultados de este modelo en un entorno de animación profesional como *Blender*. Con potencial aplicación en el campo de la robótica, se presenta un modelo de desarrollo artificial, inspirado en las gramáticas de grafos reguladas por cadenas, aplicado a la robótica evolutiva y de desarrollo para la búsqueda de formas y comportamientos creativos de agentes virtuales escaladores. Los resultados de las simulaciones confirman la adecuación del uso del formalismo de las gramáticas de grafos reguladas por cadenas, que implementa una codificación indirecta, como sistema de desarrollo de estructuras ingenieriles. Por último, se presenta un modelo de aplicación de la ingeniería embriomórfica al desarrollo de sistemas auto-organizados artificiales, del cual se han derivado dos publicaciones [55], [54]. Este modelo propone principios para el desarrollo de agentes multicelulares auto-organizados capaces de crear morfologías precisas y funcionales de una manera puramente endógena; con aplicaciones potenciales en robótica colectiva y animación.



## Agradecimientos

Me gustaría comenzar mostrando mi agradecimiento a mi tutor y director de tesis, Francisco J. Vico, sobre todo por haberme brindado la oportunidad de embarcarme en esta estimulante aventura que resulta el mundo de la ciencia. Han sido varios años de travesía en los cuales realmente he disfrutado satisfaciendo mi curiosidad e intentando aportar mi humilde granito de arena al campo científico. Desde luego, todo ello no hubiera sido posible sin ti, gracias Paco. Tampoco quiero dejar pasar la oportunidad de agradecer al Dr. Rene Doursat el hecho de acogerme durante mi estancia en el Instituto de Sistemas Complejos de París, momento desde el cuál surgió una enriquecedora colaboración entre ambos a nivel científico. Gracias también al programa de becas FPI de la Junta de Andalucía, el cual ha financiado mis investigaciones durante cuatro años.

Como estudiante de doctorado a cargo de Francisco J. Vico, he realizado mi labor investigadora dentro del Grupo de Estudios en Biomimética (GEB), el cual pertenece al grupo de Inteligencia Computacional y Análisis de Imágenes (ICAI), liderado por José Muñoz Pérez, a quién me gustaría agradecer su trato amable en los momentos que he coincidido con él. Este agradecimiento es extensible al resto de miembros del ICAI y del departamento de lenguajes y ciencias de la computación de la Universidad de Málaga.

Gran parte de culpa de mi agradable experiencia la tienen mis compañeros de laboratorio, los que aún están y los que ya se marcharon. Personas con las que he compartido grandes momentos e interesantes charlas. Gracias a José David Fernández, por sus constantes y sabios consejos que han permitido que este camino, en ocasiones tortuoso, sea más llevadero. También me gustaría darle las gracias a Miguel Carmona por compartir sus iluminadores y refrescantes conocimientos conmigo. Tampoco puedo olvidarme del siempre perspicaz José David Gutiérrez

“Guti” y de David Albarracín, tan amante de la tecnología. Gracias también a José Serrano y Juan Carlos Moya por echarme una mano con sus amplios conocimientos musicales; a Francisco Osuna, por su apoyo administrativo y amabilidad; a Julio Pérez, por su soporte técnico durante estos años; a Daniel Lobo y Gema Martín, por haber compartido su entusiasmo por la ciencia en mis comienzos. También agradecer a todos los demás que han pasado por aquí de los cuáles guardo un buen recuerdo: Fran Moreno, Ellen Pearce, Salva Burrezo, Vicente Canteli, Juan Travesero, Cristóbal Carnero, Alejandro Villegas, Rafael De Vicente y Fran Palma.

Por supuesto sería imposible no hacer una mención muy especial a mi familia, el motor de mi vida. Gracias a mis padres, Puri y Eugenio, por todo lo que me han dado y por sacrificarse para procurar que nunca me faltase de nada. Gracias a mis hermanos David, Susana y Raúl por tantos momentos de cariño y al resto de mi familia por su apoyo. Por último, agradecer a Manuela por ser mi fiel compañera y dejarme viajar a su lado en el vagón de esta montaña rusa que es la vida.

Gracias a todos.

# Índice general

<b>Resumen</b>	<b>vii</b>
<b>Agradecimientos</b>	<b>ix</b>
<b>Índice de Figuras</b>	<b>xv</b>
<b>Índice de Tablas</b>	<b>xix</b>
<b>1. Introducción</b>	<b>1</b>
1.1. Algoritmos evolutivos .....	4
1.2. Biología evolutiva del desarrollo .....	5
1.3. Ingeniería morfogénica y embriomórfica .....	8
1.4. Contribuciones .....	10
1.5. Organización del contenido .....	12
<b>2. <i>Melomics</i>: Un nuevo enfoque en composición algorítmica inspirado en <i>evo-devo</i></b>	<b>15</b>
2.1. Introducción y trabajo previo relacionado .....	16
2.1.1. Contribuciones de la inteligencia artificial a la composición algorítmica .	16
2.2. <i>Melomics</i> : un nuevo enfoque dentro del campo de la creatividad computacional.....	20
2.2.1. Ordenadores compositores: <i>Iamus</i> y <i>Melomics109</i> .....	22
2.2.2. Del laboratorio al mercado: hitos y logros alcanzados por <i>Melomics</i> .....	25
2.3. Validación de la tecnología <i>Melomics</i> como sistema de generación musical ...	30
2.3.1. Participantes.....	31
2.3.2. Materiales .....	32
2.3.3. Procedimiento .....	33
2.3.4. Resultados .....	35

2.4. Validación del uso potencial de <i>Melomics</i> como sistema de tratamiento terapéutico.....	36
2.4.1. La prueba de punción cutánea ( <i>SPT</i> ) y el uso de la música en el ámbito hospitalario.....	37
2.4.2. Participantes .....	38
2.4.3. Materiales .....	39
2.4.4. Procedimiento .....	41
2.4.5. Resultados.....	42
2.5. Conclusiones .....	43
<b>3. Evolución de avatares animados mediante física como soporte creativo para la sincronización de personajes y música en escenas de animación</b>	<b>47</b>
3.1. Introducción y trabajo previo relacionado.....	48
3.1.1. Animación basada en física .....	49
3.1.2. Sincronización de personajes y música en escenas de animación .....	50
3.2. Algoritmo de detección de ritmo musical .....	52
3.3. Descripción del modelo.....	54
3.3.1. Modelado del avatar .....	55
3.3.2. Física del modelo .....	57
3.3.3. Evolución de avatares mediante función de bondad interactiva.....	58
3.4. Herramienta de soporte creativo para animación de personajes.....	63
3.5. Conclusiones .....	66
<b>4. Desarrollo artificial aplicado a búsqueda de forma y comportamiento creativos de agentes escaladores</b>	<b>69</b>
4.1. Introducción y trabajo relacionado. ....	70
4.1.1. Las gramáticas de grafos reguladas por cadenas.....	70
4.1.2. Evolución y desarrollo de morfología y control en agentes .....	71
4.1.3. Aplicaciones de los robots escaladores.....	74

4.2. Descripción del modelo .....	75
4.2.1. Proceso de desarrollo artificial .....	75
4.2.2. Variación del modelo: dominio de expresión heredado variable .....	85
4.2.3 Física del modelo .....	87
4.2.4. Algoritmo genético .....	88
4.3. Resultados obtenidos en las simulaciones .....	90
4.4. Conclusiones y discusión .....	98
<b>5. El desarrollo biológico como inspiración para la creación de agentes multicelulares</b> .....	<b>101</b>
5.1. Introducción y trabajo previo relacionado .....	102
5.1.1. Ingeniería morfogénica .....	103
5.1.2. Robótica evo-devo y soft-bodied .....	109
5.1.3. Animación basada en <i>soft-bodies</i> .....	111
5.2. Arquitectura modular mediante desarrollo programable en 3D .....	112
5.2.1 El crecimiento y desarrollo de un enjambre 3D homogéneo (SA) .....	114
5.2.2. La propagación de información posicional en un enjambre 3D ( <i>PF1</i> ) .....	115
5.2.3. Diferenciación programada en un enjambre 3D ( <i>PF2</i> ) .....	120
5.2.4. Crecimiento y creación de patrones simultáneos (SA + PF) en un enjambre 3D .....	121
5.2.5. Patrón recursivo modular ( $PF^K$ ) y crecimiento anisotrópico ( $SA^K$ ) en un enjambre 3D .....	122
5.3. Comportamiento de las arquitecturas funcionales en un entorno físico .....	125
5.3.1. Diferenciación mecánica en huesos y músculos .....	126
5.3.2. Animación mediante contracción muscular .....	127
5.4. Función a partir de la estructura, estructura a partir del desarrollo: Un proceso evolutivo doblemente indirecto.....	129

5.4.1. Polimorfismo Evolutivo: variando el genotipo .....	129
5.4.2. Polifenismo desarrollado: variando el fenotipo .....	132
5.5. Conclusiones y discusiones .....	134
<b>6. Descripción de contenidos y conclusiones</b>	<b>137</b>
6.1. Descripción de contenidos .....	137
6.2. Conclusiones .....	140
<b>Apéndice A: Detalles de implementación hardware y software</b>	<b>145</b>
A.1. Configuración del clúster de computación.....	145
A.2. Implementación de los modelos computacionales .....	147
A.3. Renderización de animaciones de personajes en <i>Blender</i> .....	149
<b>Apéndice B: Formularios y documentos utilizados en los experimentos realizados</b>	<b>155</b>
B.1. Experimento para estudiar la comparación entre la música generada por <i>Melomics</i> y la creada por seres humanos.....	155
B.2. Experimento para validar el uso potencial de <i>Melomics</i> como sistema de tratamiento terapéutico .....	163
<b>Apéndice C: Publicaciones derivadas de la tesis doctoral</b>	<b>167</b>
<b>Bibliografía</b>	<b>169</b>

## Índice de Figuras

Figura 2.1: Sistemas informáticos que implementan la tecnología <i>Melomics</i> .....	23
Figura 2.2: Ejemplos de partituras generadas por <i>Iamus</i> .....	23
Figura 2.3: <i>Iamus</i> , primer disco compuesto íntegramente por el computador <i>Iamus</i> e interpretado por la Orquesta Sinfónica de Londres.....	26
Figura 2.4: Página principal del repositorio creado por <i>Melomics109</i> que cuenta con piezas musicales de música popular abarcando todos los estilos esenciales... ..	27
Figura 2.5: Aplicaciones móviles que ofrecen música por <i>streaming</i> usando la <i>API</i> de acceso a los contenidos del repositorio.....	29
Figura 2.6: <i>Omusic</i> , primer disco compuesto e interpretado íntegramente por el computador <i>Melomics109</i> .. ..	30
Figura 2.7: Imágenes de la realización del experimento llevado a cabo para comparar la música compuesta por <i>Melomics</i> y la compuesta por humano.....	33
Figura 2.8: Descripción general del sistema terapéutico <i>M<sup>3</sup></i> .....	40
Figura 2.9: Escala Visual Analógica.....	41
Figura 2.10: Valor de la mediana de la EVA en el grupo experimental y el grupo control. ....	42
Figura 2.11: Porcentaje de niños que reportaron presencia de dolor. ....	43
Figura 3.1: Algoritmo de detección de <i>beats</i> para una señal de entrada.. ..	54
Figura 3.2: Representación de un avatar usando el motor de física <i>ODE</i> .....	56
Figura 3.3: Representación del genotipo de un individuo. ....	60
Figura 3.4: Asignación interactiva del valor de bondad de los individuos.. ..	62
Figura 3.5: Esquema del modelo y herramienta propuestos para la animación automática de coreografías.. ..	64
Figura 3.6: Visualización del avatar en tres estados del proceso de creación de la animación.. ..	65

Figura 3.7: Inclusión de un personaje evolucionado para bailar ballet en una escena. ....	66
Figura 4.1: Correspondencia entre grafo desarrollado mediante una gramática regulada por cadenas y organismo funcional que representa.....	77
Figura 4.2: Regla de producción $R$ .....	80
Figura 4.3: Regla de producción $M$ .....	81
Figura 4.4: Regla de producción $S$ .....	83
Figura 4.5: Proceso de desarrollo de un organismo simple desarrollado a partir del genoma.....	84
Figura 4.6: Comparación de comportamiento del gen $R$ en la estrategia de fomento de simetría bilateral y en la estrategia de no fomento de simetría.....	86
Figura 4.7: Promedio del valor de bondad del mejor individuo, medio y peor para las 25 ejecuciones llevadas a cabo para cada versión del modelo. ....	91
Figura 4.8: Distintas etapas del proceso de desarrollo de tres de los mejores organismos obtenidos con la estrategia de fomento de simetría corporal.....	92
Figura 4.9: Distintas etapas del proceso de desarrollo de tres de los mejores organismos obtenidos con la estrategia de no fomentar la simetría corporal.....	93
Figura 4.10: Simulación del individuo con mejor valor de bondad desarrollado usando la estrategia de fomento de la simetría.....	96
Figura 4.11: Simulación del individuo con mejor valor de bondad desarrollado usando la estrategia de no fomento de la simetría.....	97
Figura 5.1: Vista previa de los organismos <i>MapDevo3D</i> y escenarios.....	104
Figura 5.2: Las cuatro categorías de la ingeniería morfogenética, cada una de ellas ilustrada con dos ejemplos de sistemas robóticos, físicamente realizados o simulados.....	105
Figura 5.3: Visión general de los mecanismos esenciales de <i>MapDevo3D</i> .....	113
Figura 5.4: Crecimiento de un enjambre 3D homogéneo ( $SA$ ).....	116
Figura 5.5: Propagación auto-organizada de la información posicional ( $PF1$ ).....	119



Figura 5.6: Formación de patrones mediante diferenciación programada ( <i>PF2</i> ) ....	121
Figura 5.7: Crecimiento y patrón simultáneos ( <i>SA + PF</i> ).....	122
Figura 5.8: Crecimiento modular y la creación de patrones ( $SA^k + PF^k$ ).....	124
Figura 5.9: Desarrollo de una extremidad individual. ....	124
Figura 5.10: Dinámicas y diferenciación estructural.....	128
Figura 5.11: Ilustración de varios tipos de adaptación fenotípica en un modelo de crecimiento programable.. ....	130
Figura 5.12: Exploración paramétrica simple de un organismo <i>MapDevo3D</i> en una tarea de subir escaleras.. ....	131
Figura 5.13: Diferentes casos de especiación en <i>MapDevo3D</i> .. ....	133
Figura A.1: Contenido del fichero <i>cuerpo.txt</i> .. ....	149
Figura A.2: Extracto del <i>script importArmadura.py</i> . ....	150
Figura A.3: Importación de la armadura en <i>Blender</i> usando el <i>script importArmadura.py</i> .....	151
Figura A.4: Importación automática de la animación en la armadura previamente creada.....	152
Figura A.5: Importación de tema musical en la animación mediante el Editor de Secuencias de Video disponible en <i>Blender</i> . ....	153
Figura A.6: Proceso de <i>binding</i> mediante el cual se incorpora una textura a la armadura.....	153
Figura A.7: Animación de armadura y textura tras el proceso de <i>binding</i> .. ....	154
Figura A.8: Inclusión del personaje animado en una escena.....	154



# Índice de Tablas

Tabla 2.1: Distribución de los participantes en dos grupos A y B. ....	34
Tabla 2.2: Resultados del experimento realizado para comparar la música generada por <i>Melomics</i> con la música creada por un humano. ....	36
Tabla A.1: Visión global del clúster .....	146



# Capítulo 1

## Introducción

La compleja cuestión acerca de si una computadora puede tener un comportamiento creativo ha despertado un creciente interés en las últimas décadas [147]. Es un hecho evidente que las computadoras han superado la capacidad humana en múltiples dominios tales como jugar al ajedrez, la clasificación musical o el procesamiento de información. Sin embargo, intentar acercarse, o igualar la creatividad humana, sigue suponiendo un reto para las computadoras, siendo considerada la capacidad creativa humana como un factor fundamental en su éxito intelectual, la cual le diferencia del resto de seres. Este hecho permite plantear la cuestión acerca de si los humanos poseen un cierto sentido especial, del cual surge la creatividad, que no puede ser transcrito a un algoritmo y, por lo tanto, no puede ser implementado por una computadora. Como respuesta a esto, la creatividad computacional surge como un campo dentro de la inteligencia artificial, que se encarga del estudio y desarrollo de sistemas hardware y software que sean capaces de exhibir un comportamiento creativo propio del ser humano. A pesar de la visión pesimista planteada anteriormente desde el punto de vista de las aspiraciones y capacidades de la creatividad computacional, desde hace décadas se han propuesto diversos modelos computacionales para resolver tareas de naturaleza creativa, tales como el desarrollo de conceptos matemáticos innovadores [107], [38], [111], la composición musical [126], [40], la escritura de historias [123], [177] y poemas [70], la creación de chistes [148], [166], retratos abstractos [68], pinturas representacionales [22], vídeo arte [31] y diseños de ingeniería creativos [137]. Aunque en su origen el término creatividad suele estar asociado a la expresión puramente artística, también puede ser extendido a otros campos tales como el diseño de formas innovadoras en la industria o la creación de sistemas robóticos con

comportamientos eficientes. Cabe reseñar que, además de buscar que las computadoras sean capaces de realizar procesos creativos por sí mismas, uno de los objetivos que persigue la creatividad computacional es que las computadoras sirvan de soporte a los humanos y actúen como herramientas que les ayuden en sus procesos creativos.

Por otra parte, la observación de la naturaleza ha sido una de las principales fuentes de inspiración para la propuesta de novedosas soluciones creativas en diferentes áreas y contextos. En este sentido, dentro de la inteligencia artificial, el paradigma bioinspirado de la computación evolutiva aborda la resolución de problemas mediante técnicas basadas en poblaciones de individuos, siendo capaces de combinar buenas soluciones con el objetivo de obtener otras mejores. La computación evolutiva engloba un amplio grupo de técnicas, básicamente divididas en dos ramas: los algoritmos evolutivos y la inteligencia de enjambres. La evolución natural representa un ejemplo extremo de proceso creativo, ya que, durante millones de años, la evolución de los seres vivos ha hecho posible la emergencia de un elevado número de diseños biológicos [21]. Por este motivo, los algoritmos evolutivos han sido empleados ampliamente en procesos creativos, incluyendo, diseños de ingeniería, diseños arquitectónicos y arte [43].

Por definición, la creatividad requiere amplias dosis de innovación y diversidad. En el campo de la biología, la síntesis evolutiva moderna [92] presenta dificultades para explicar el surgimiento de la multitud de formas y comportamientos innovadores de los seres vivos que pueblan (y han poblado) la Tierra. Esta deficiencia está causada principalmente por la omisión del proceso de desarrollo [130], en el que una sola célula (cigoto) se transforma en un complejo organismo con miles de millones de células. Por este motivo, el campo de la biología evolutiva del desarrollo, conocida como *evo-devo*, ha emergido para reclamar la importancia del proceso de desarrollo regulado por el genoma en el surgimiento de innovación y diversidad [26]; y su incorporación como componente clave para la evolución de una gran diversidad de

## 1.1. Algoritmos evolutivos

---

comportamientos y diseños estructurales innovadores, características ambas muy apreciadas en los procesos creativos. En el campo de la computación, la biología evolutiva del desarrollo ha inspirado dos disciplinas: el desarrollo artificial, que incorpora el proceso de desarrollo en los algoritmos evolutivos mediante codificación indirecta del esquema genotipo-fenotipo; y la ingeniería embriomórfica la cual, imitando el proceso de desarrollo biológico, persigue el desarrollo de morfologías y comportamientos complejos artificiales mediante la agregación descentralizada y la auto-organización de una gran cantidad de pequeños agentes.

En línea con la compleja cuestión planteada inicialmente sobre la capacidad de las computadoras para realizar tareas de índole creativo, este trabajo de tesis explora la aplicación de diferentes disciplinas bioinspiradas, concretamente los algoritmos evolutivos, el desarrollo artificial y la ingeniería embriomórfica, de forma individual o combinada, para la generación de productos creativos. Para ello se proponen modelos computacionales que actúan de soporte a la creatividad humana, o que exhiben comportamiento creativo de forma independiente, y cuyas soluciones son aplicables en contextos tan diversos como la composición algorítmica, la medicina, la robótica y la animación por computador.

Una vez presentada la línea de investigación de esta tesis, el resto del capítulo se dedicará a realizar una introducción a las disciplinas bioinspiradas relacionadas, con el objetivo de poner en contexto cada uno de los modelos presentados. De este modo, la sección 1.1 presenta una visión general de los algoritmos evolutivos como herramientas de búsqueda y optimización. La sección 1.2 introduce los fundamentos de la biología evolutiva del desarrollo y sus implicaciones en la computación evolutiva. La disciplina de la ingeniería morfogénica y su caso particular, la ingeniería embriomórfica, la cual toma su inspiración de la biología evolutiva del desarrollo, son introducidas en la sección 1.3. La sección 1.4 presenta un resumen de las contribuciones de este trabajo de tesis. Finalmente, una breve exposición del resto del contenido de este documento de tesis es presentada en la sección 1.5.

## 1.1. Algoritmos evolutivos

La evolución de los seres vivos presente en la naturaleza, ha sido fuente de inspiración para una serie de métodos de búsqueda y optimización denominados algoritmos evolutivos los cuales, junto con la inteligencia de enjambre, conforman el campo de la computación evolutiva [47]. Los algoritmos evolutivos han sido estudiados en profundidad en las últimas décadas, y se han aplicado a muchos dominios de problemas diferentes [3], [46], [182], [149], especialmente problemas computacionales que requieran una búsqueda de soluciones dentro de un espacio de búsqueda muy amplio [127].

La mayoría de los algoritmos evolutivos siguen un patrón común: un conjunto o población cambiante de soluciones candidatas es sometido a un ciclo repetitivo de evaluación, selección y reproducción con variación. El primer paso en el algoritmo consiste en generar las soluciones candidatas que forman la población inicial, generalmente de forma aleatoria. Cada candidato se evalúa a continuación, utilizando una función de bondad que mide su calidad como solución del problema. Después de la fase de evaluación, la selección se lleva a cabo: una nueva población de soluciones candidatas es generada a partir de copias de las soluciones de la población anterior; cada solución candidata se copia un número de veces probabilísticamente proporcional a su valor de bondad. Este paso reduce la diversidad de la población, eliminando probabilísticamente los individuos con peor bondad. Finalmente, la reproducción produce nuevas soluciones candidatas variando aleatoriamente los individuos de la población actual para aumentar la variación mediante operadores de mutación o recombinación. Estos pasos (evaluación, selección y reproducción con la variación) se aplican de forma iterativa evolucionando los individuos de la población y promocionando las mejores soluciones. De esta forma, el algoritmo evolutivo es capaz de sesgar la exploración



del espacio de soluciones, con respecto a la búsqueda aleatoria, invirtiendo mayor tiempo en aquellas áreas que parezcan más prometedoras [185].

Dentro de los algoritmos evolutivos, existen varias técnicas que pueden usar diferentes conjuntos de reglas de selección, operadores de variación y codificación de la solución. Las variedades más conocidas son los algoritmos genéticos [88], las estrategias evolutivas [8], la programación evolutiva [66] y la programación genética [105]. De entre estas técnicas, el algoritmo genético es el que mejor representa a la evolución biológica, ya que es el único que distingue entre genotipo y fenotipo. En un algoritmo genético la forma codificada de una solución candidata corresponde al genotipo, mientras que el fenotipo es la traducción de esa forma codificada en una solución al problema. El esquema de codificación del genotipo, es un aspecto muy importante en la implementación de un algoritmo genético [102]. En su formulación original, los algoritmos genéticos han usado métodos de codificación directa, siendo los individuos representados mediante una fila de datos vinculados de forma explícita a sus propiedades. Sin embargo, tal y como se presenta en la siguiente sección existe otra forma alternativa de implementar esta codificación mediante métodos de codificación indirecta (o implícitos). De hecho, los modelos presentados en los capítulos 2, 4 y 5 utilizan algoritmos genéticos con codificación indirecta para hacer evolucionar diversas estructuras, mientras que el modelo propuesto en el capítulo 3, utiliza una versión clásica del algoritmo genético con codificación directa.

## 1.2. Biología evolutiva del desarrollo

La evolución biológica es uno de los principales mecanismos que ha contribuido a la diversidad y complejidad de las formas de vida. Como se ha mencionado en la anterior sección, en ciencias de la computación, los algoritmos evolutivos representan un tipo de técnicas metaheurísticas inspiradas en la biología evolutiva

[15]. Sin embargo, los algoritmos evolutivos clásicos tienden a mostrar problemas de escalabilidad (el rendimiento se degrada significativamente a medida que el tamaño del problema aumenta) y de estructura de la solución (las soluciones generadas por el algoritmo tienden a ser no estructuradas, difíciles de adaptar y frágiles [91]). Uno de los factores más importantes en estos problemas es el uso de codificación directa: los algoritmos evolutivos clásicos usan genotipos (representaciones de las soluciones) que se convierten en fenotipos (las soluciones en sí mismas) de una manera directa y sencilla. Cuando un algoritmo utiliza codificación directa, cada parte de la solución se asigna a una parte de la representación. Como resultado, los genotipos pueden resultar demasiado grandes para que la optimización evolutiva llegue a ser práctica, dando lugar a que las diferentes partes del genotipo evolucionen de forma no coordinada, induciendo los problemas de escalabilidad y fragilidad mencionados anteriormente.

El uso de codificación directa puede ser considerado en desuso. En la biología evolutiva contemporánea, el desarrollo (el proceso que transforma un cigoto en un organismo multicelular complejo) se sabe que juega un papel fundamental en los procesos evolutivos [26]. Desafortunadamente, desde que los algoritmos evolutivos fueron propuestos, el papel del proceso de desarrollo no ha sido tenido en cuenta durante decenios en parte debido a que su implementación aumentaba la complejidad computacional del algoritmo. Esto, unido a la limitada capacidad de los primeros computadores, dio lugar a la adopción generalizada del uso de la codificación directa en los algoritmos evolutivos.

A medida que pasaba el tiempo, la importancia del desarrollo en el contexto de la biología evolutiva comenzó a ser mejor entendido, y esto dio origen a una rama específica del pensamiento evolutivo: la biología evolutiva del desarrollo (*evo-devo*). Los procesos de desarrollo se pueden describir como coreografías auto-organizadas de eventos precisamente cronometrados, con células que se dividen y se organizan por sí mismas en capas de tejidos que se pliegan en formas complejas, resultando en

la formación de un organismo multicelular a partir de un cigoto. En *evo-devo*, los cambios evolutivos son interpretados como pequeñas mutaciones en el genoma del organismo, que modulan sus procesos de desarrollo en complejas y orquestadas formas, dando como resultado la alteración de formas y características novedosas.

Conforme el campo de la *evo-devo* fue madurando dentro del contexto de la biología evolutiva, inspiró un campo análogo en el contexto de la computación evolutiva: el desarrollo artificial. En oposición a las codificaciones directas de los algoritmos evolutivos clásicos, el desarrollo artificial usa codificaciones indirectas, es decir, abstracciones formales de los procesos de desarrollo que constituyen una correspondencia compleja entre genotipo y fenotipo. Usando una efectiva codificación indirecta, un genotipo sencillo puede potencialmente especificar un fenotipo complejo, solucionando el problema de escalabilidad previamente mencionado. Adicionalmente, un pequeño cambio en el genotipo puede potencialmente dar lugar a muchos cambios coordinados en el fenotipo, lo que ayuda a mitigar el problema de la fragilidad.

Debido a estas características, los algoritmos evolutivos con codificaciones indirectas bien diseñadas pueden obtener soluciones complejas, y generar intrincadas variaciones de estas soluciones. Este hecho hace que estos algoritmos hayan irrumpido en muchos campos que tradicionalmente se han basado exclusivamente en la experiencia humana, ya que requieren procesos creativos, como el diseño industrial o las artes. En otras palabras, pueden ser usados para automatizar (hasta cierto punto) las tareas que exigen creatividad, proponiendo diferentes variaciones de las soluciones existentes, que evolucionan hacia las metas de diseño deseadas, asemejándose a una forma automatizada de *brainstorming*.

Algunos ejemplos notables de las aplicaciones en la vida real del desarrollo artificial han aparecido en los últimos años, tales como el conjunto de antenas diseñadas para adaptarse a los requisitos técnicos de los satélites de una misión espacial de la NASA

[90]; el diseño de fibras ópticas micro-estructuradas [117]; la generación automática de juegos de mesa [28]; y la animación automática de personajes [112].

Los modelos propuestos en los capítulos 2, 4 y 5 aplican este paradigma de desarrollo artificial a diferentes dominios de la creatividad computacional como pueden ser la composición musical, el diseño de forma y comportamiento de sistemas robóticos y la animación de personajes.

### 1.3. Ingeniería morfogénica y embriomórfica

Hoy en día los ingenieros se debaten entre un control total sobre el diseño y el deseo de crear dispositivos autónomos: por una parte, quieren el pleno control del diseño de sus artefactos, mientras que por otra desean que estos sean más adaptables, inteligentes o que muestren comportamientos creativos de forma endógena. Por un lado, mientras que todavía está muy arraigado el paradigma programación-depuración-actualización en el diseño de nuestros sistemas informáticos y robóticos más sofisticados, la demanda de innovación y creatividad ha creado un aumento en el tamaño de estos sistemas y su complejidad. En este contexto, el enfoque tradicional de una planificación *top-down* rígida es insostenible. Por otro lado, los sistemas complejos naturales, grandes conjuntos de elementos que interactúan a nivel local y que producen comportamientos colectivos no triviales, ofrecen una alternativa viable y pueden suponer una fuente de ideas innovadoras.

La ingeniería morfogénica, el tema principal de un libro reciente [53], se refiere al diseño, o más bien metadiseño, de las capacidades de auto-organización de los elementos de los sistemas complejos para su aplicación en arquitecturas funcionales.

Por lo general, los fenómenos naturales de formación de patrones espontáneos son aleatorios y repetitivos [10], mientras que, en el extremo opuesto del espectro, los artefactos y los dispositivos elaborados son el producto determinista del diseño

### 1.3. Ingeniería morfogénica y embriomórfica

---

humano. Sin embargo, los organismos biológicos multicelulares (y, en cierta medida, las construcciones colectivas de insectos) son notables ejemplos de sistemas complejos que son totalmente auto-organizados y arquitectónicos. En consecuencia, el objetivo de la ingeniería morfogénica es establecer un nuevo campo de investigación para explorar la intersección entre estos dos dominios tradicionalmente desconectados, es decir, el modelado y la implementación de sistemas auto-organizados [56]. La ingeniería morfogénica pone especial énfasis en las habilidades de computación y capacidad de programación de los sistemas auto-organizados, propiedades que, a menudo, no se les da el protagonismo que merecen en metodologías ingenieriles.

La ingeniería embriomórfica [52], [51], [55], un caso particular de la ingeniería morfogénica, toma su inspiración directamente de la *evo-devo* para crear nuevas arquitecturas de red, hardware o software mediante la agregación descentralizada y la auto-organización de una gran cantidad de pequeños agentes, o células, explorando esta auto-organización descentralizada en morfologías y comportamientos complejos artificiales. En esencia, la ingeniería embriomórfica combina tres principios clave de la embriogénesis multicelular: (1) la difusión de gradiente químico, que proporciona información sobre la posición de las células; (2) las redes reguladoras genéticas, que provocan la diferenciación de células en tipos, creando así patrones; y (3) la división celular, que impone restricciones estructurales, creando así nuevas formas. En el capítulo 5, se muestra el potencial de aplicación de la ingeniería embriomórfica a la robótica colectiva y la animación *soft-bodied*, a través de un modelo abstracto 3D para el desarrollo de organismos multicelulares artificiales.

## 1.4. Contribuciones

El carácter interdisciplinar de este trabajo de tesis, aporta varios resultados derivados del estudio de nuevos modelos computacionales bioinspirados, y su aplicación como herramientas creativas en campos tan diversos como la composición musical, la medicina, la animación por computador y la robótica. En lo que sigue, se resumen brevemente las principales contribuciones científicas derivadas de esta tesis, que serán más ampliamente discutidas en posteriores capítulos.

En el campo de la composición musical, se presenta *Melomics* un nuevo enfoque en creatividad computacional, que aplica computación evolutiva y desarrollo artificial para favorecer la evolución de estructuras musicales complejas (capítulo 2). Este enfoque proporciona nuevas estrategias para la codificación de piezas musicales y plantea varias áreas de aplicación industrial dentro de la composición musical. La calidad compositiva de la música generada por *Melomics* es contrastada mediante los resultados de un experimento, que demuestran que la música compuesta por *Melomics* resulta indistinguible de la compuesta por un humano, situándose a un nivel similar que ésta en lo que se refiere a generación de emociones y evocación de representaciones mentales. En el campo de la medicina, se presentan los resultados de un experimento que ponen de manifiesto los beneficios de la aplicación de  $M^3$  (abreviatura de *Melomics Music Medicine*), un subsistema de *Melomics*, como metodología terapéutica, lo que contribuye a plantear el origen de una amplia e innovadora familia de herramientas terapéuticas.

En el campo de la animación por computador, se propone una herramienta de creatividad computacional para la animación de personajes en base a música (capítulo 3). Esta herramienta está basada en un modelo que evoluciona avatares animados mediante física y supone una fuente generadora de ideas para animadores aficionados o profesionales, proporcionándoles de forma automática la animación de un personaje que puede ser utilizada directamente o editada con un software de

animación como *Blender*. Por un lado, al usar un enfoque basado en física se obtienen soluciones atractivas visualmente; mientras que, por otro, la ausencia de sesgo y de conocimiento previo de la computadora, permite explorar y obtener soluciones (en este caso coreografías de baile) que un diseñador humano difícilmente podría concebir.

Las contribuciones al campo de la robótica, por otra parte, derivan de un modelo de desarrollo artificial, inspirado en el formalismo de las gramáticas de grafos reguladas por cadenas, aplicado a la robótica evolutiva y de desarrollo, para la búsqueda de formas y comportamientos creativos de agentes virtuales escaladores (capítulo 4). Este modelo explota las capacidades creativas y de optimización de los algoritmos evolutivos con codificación indirecta para el diseño, tanto de forma como de control y comportamiento, de estructuras de carácter ingenieril, concretamente agentes virtuales que se enfrentan al reto evolutivo de escalar por una pendiente. Aplicado a la robótica, el modelo propuesto puede actuar como soporte de apoyo creativo al diseñador o ingeniero, inspirando y proporcionando soluciones innovadoras en el diseño de un robot.

Con potenciales aplicaciones tanto en el campo de la robótica y la animación *soft-bodied*, se presenta también un modelo de ingeniería embriomórfica, inspirado en el desarrollo biológico, para la creación de agentes multicelulares artificiales (capítulo 5). Este modelo contribuye al diseño de nuevos sistemas auto-organizados capaces de complementar, o incluso sustituir, a los diseñadores o ingenieros mediante grandes colectividades descentralizadas de agentes. Para ello se usan principios para el metadiseño de organismos auto-organizados capaces de crear morfologías precisas e innovadoras de una manera puramente endógena, bajo el control de un programa genómico.

Finalmente, como aportación específica al campo de la animación basada en física, el modelo presentado en el capítulo 3, contribuye con un nuevo enfoque para la

aplicación de esta disciplina en la sincronización de personajes y música en escenas de animación. Por otro lado, el modelo propuesto en el capítulo 5, aporta un método para el desarrollo de criaturas en animación *soft-bodied*, que interactúan con entornos virtuales físicamente realistas. Así mismo, el modelo descrito en el capítulo 4, propone un método que puede ser trasladado al desarrollo de criaturas que combinan características propias de la animación *soft-bodied* y *rigid-bodied* y que, al igual que en el caso anterior, también tienen la capacidad de interactuar en un entorno virtual sujeto a las leyes de la física.

## 1.5. Organización del contenido

El resto de este trabajo de tesis se organiza de la siguiente manera. El capítulo 2 presenta *Melomics*, un nuevo enfoque en creatividad computacional basado en *evo-devo* aplicado a la composición algorítmica. Su comparación con la música compuesta por humanos y la aplicación de la tecnología derivada a los sistemas terapéuticos, son estudiados mediante sendos experimentos controlados llevados a cabo. El capítulo 3, presenta un modelo evolutivo de avatares animados mediante física como soporte creativo para la sincronización de personajes y música en escenas de animación. Se propone una herramienta para la importación automática de los resultados de este modelo en un entorno de animación como *Blender*. El capítulo 4, propone un modelo de desarrollo artificial aplicado a la robótica evolutiva y de desarrollo para la búsqueda de formas y comportamientos creativos de robots escaladores. El capítulo 5, presenta un modelo de ingeniería embriomórfica, que utiliza el desarrollo biológico como inspiración para la creación de agentes multicelulares con aplicaciones en robótica colectiva y animación. El capítulo 6, presenta un resumen del trabajo de tesis, junto con las principales conclusiones derivadas del mismo.



## 1.5. Organización del contenido

---

Finalmente, diversos apéndices incluidos al final de esta memoria proporcionan información y documentación complementaria sobre diversos aspectos de la tesis doctoral: en el apéndice A, se detallan las características de configuración hardware del clúster de computación utilizado en las simulaciones de algunos de los modelos computacionales presentados, así como se aportan detalles sobre la implementación software de estos modelos. El apéndice B, recoge los modelos de formularios y cuestionarios utilizados en los experimentos realizados en el capítulo 2. Por último, en el apéndice C, se incluye el listado de publicaciones generadas como resultado de esta tesis doctoral.



## Capítulo 2

### ***Melomics*: Un nuevo enfoque en composición algorítmica inspirado en *evo-devo*.**

En este capítulo se presenta *Melomics*, un nuevo enfoque dentro del campo de la creatividad computacional, que se centra en la composición algorítmica, y tiene como objetivo la automatización completa del proceso de composición de la música profesional. *Melomics* sigue un enfoque *evo-devo*, ya que combina un algoritmo evolutivo con un esquema de codificación indirecta para favorecer la evolución de estructuras musicales complejas a partir de genomas relativamente simples. *Melomics* ha sido considerado como un hito en la investigación en inteligencia artificial [9], [17] y plantea varias áreas de aplicación industrial dentro de la composición musical, tales como la posibilidad de disponer de un repositorio ilimitado de piezas musicales libres de derechos de autor, acceder a nuevas herramientas para la manipulación de la música, o poner en práctica la “computación sentiente” para ofrecer música adaptada a las necesidades del usuario. Estas aplicaciones abren así la puerta a la creación de nuevos modelos de negocio y aplicaciones en diferentes campos, entre ellos el de los sistemas terapéuticos. Lejos de ofrecer una descripción detallada sobre el núcleo de la tecnología del sistema *Melomics*, el objetivo de este capítulo está orientado a realizar un recorrido sobre los principales resultados alcanzados por este sistema en el campo de la creatividad computacional, centrándose en el estudio de su comparación con la música compuesta por humanos; y sus aplicaciones en el ámbito terapéutico.

Este capítulo está organizado de la siguiente manera. La sección 2.1 incluye una breve revisión de las diferentes técnicas de inteligencia artificial presentes en la literatura aplicadas a la composición algorítmica. La sección 2.2 introduce *Melomics*

como un nuevo enfoque de creatividad computacional aplicado a la composición algorítmica, que se basa en los principios de la biología evolutiva del desarrollo. Además se presentan los hitos más importantes que han sido alcanzados por los dos supercomputadores que han aplicado esta tecnología: *Iamus* y *Melomics109*. La sección 2.3 muestra los resultados de un estudio realizado para explorar en qué medida la música generada por *Melomics* tiene la misma valencia que la compuesta e interpretada por los seres humanos. La aplicación de la tecnología *Melomics* como sistema de tratamiento terapéutico es descrita en la sección 2.4, mediante un experimento controlado llevado a cabo en un ámbito pediátrico. Por último, en la sección 2.5, se discuten las conclusiones y aportaciones que se derivan de este innovador enfoque.

## 2.1. Introducción y trabajo previo relacionado

El modelo presentado en este capítulo supone uno de los más recientes resultados exitosos en computación evolutiva, inspirándose en los principios de la biología evolutiva del desarrollo para la evolución de composiciones musicales complejas, con un coste computacional relativamente bajo. En el capítulo de introducción se ha dedicado una sección para introducir los principios y fundamentos de la biología evolutiva del desarrollo, por este motivo, el resto de esta sección se dedicará a realizar una revisión de los trabajos previos relacionados con la aplicación de técnicas de inteligencia artificial a la composición algorítmica.

### 2.1.1. Contribuciones de la inteligencia artificial a la composición algorítmica

Para alcanzar los resultados obtenidos en *Melomics*, en vez de las técnicas de inteligencia artificial tradicionales, se ha seguido un enfoque basado en desarrollo artificial con codificación indirecta. En la literatura podemos encontrar una amplia gama de técnicas basadas en inteligencia artificial que se han utilizado para la

composición algorítmica tales como los sistemas gramaticales y basados en el conocimiento, las redes neuronales artificiales y los algoritmos evolutivos. El conjunto de trabajos más significativos basados en estos métodos y técnicas han sido recogidos en varios trabajos de revisión como el llevado a cabo por Fernández y Vico en 2013 [63]. El primer resultado destacable en composición algorítmica se remonta a los orígenes de la inteligencia artificial en 1956: *Illiacc Suite* [85], una composición generada por computador mediante un sistema de reglas y cadenas de Markov, llevado a cabo como un experimento sobre los aspectos formales de la composición musical. Desde entonces, muchos investigadores y artistas han conseguido resultados notables. Iannis Xenakis, un reconocido compositor de vanguardia, utilizó de forma habitual algoritmos estocásticos para generar materia prima para sus composiciones, usando computadoras desde la década de 1960 para automatizar estos métodos [4]. En 1964 otro compositor, Koenig, no tan conocido como Xenakis, implementó un algoritmo denominado *PROJECT1* utilizando composición en serie (una teoría musical) y otras técnicas (como las cadenas de Markov) para automatizar la generación de música [4].

En cuanto al uso de sistemas gramaticales y métodos relacionados, Prusinkiewicz [138], en 1986 utilizó sistemas de Lindenmayer (más conocidos por el término en inglés *L-systems*) para la composición algorítmica usando gráficos tortuga para renderizar una imagen, que posteriormente era interpretada como una partitura musical. Otros autores han usado también un enfoque basado en *L-systems* pero sin dependencia de la interpretación gráfica, como es el caso de Morgan [129] en 2007. En cuanto al uso de gramáticas en conjunción con algoritmos evolutivos, en 1999, Thywissen, en su sistema *GeNotator* [172], evolucionaba reglas de gramáticas que eran representadas mediante genomas y evaluadas con una función de bondad interactiva (el usuario asigna el valor de bondad a la gramática). Pero quizás el modelo más conocido usando sistemas gramaticales es el proyecto *EMI* (*Experimentals in Musical Intelligence*), de David Cope [39]. Este trabajo se centra en la

emulación de estilos de diferentes compositores y ha compuesto con éxito música en los estilos de Mozart, Bach, Rachmaninov, Albinoni, Brahms, Chopin y Stravinsky. Funciona mediante la búsqueda de patrones recurrentes en varios trabajos de un compositor determinado. Cope ha plasmado el diseño de EMI en otras aplicaciones tales como SARA, ALICE [41] y *Emily Howell* [40].

Los sistemas expertos también han sido ampliamente aplicados a la composición algorítmica. Además del ya mencionado *Illiad Suite*, *CHORAL* [58] de Kemal Ebcioğlu's (1998), es un sistema experto que armoniza corales en el estilo de J.S. Bach, recibiendo una melodía y produciendo su correspondiente armonización, basándose en heurísticas y restricciones. *CHORAL* fue implementado utilizando un lenguaje de programación lógica diseñado por el autor. Un aspecto importante de este trabajo es el uso de conjuntos lógicos de primitivas para representar los diferentes puntos de vista de la música (acordes, intervalo de tiempo, melodía, etc.). Otro trabajo pionero utilizando sistemas expertos fue el realizado por Gill [71] en 1963, usando una búsqueda jerárquica con “vuelta atrás” para guiar un conjunto de reglas. En cuanto a la integración de algoritmos evolutivos y sistemas expertos, la forma más intuitiva consiste en diseñar una función de bondad en base a un conjunto de reglas del sistema. Un buen ejemplo es el trabajo de McIntyre [122] en 1994, que extrae un conjunto de reglas para la armonización de obras clásicas, y las codifica como un conjunto de funciones de puntuación. El valor de bondad resultaba de la suma ponderada de estas puntuaciones con una estructura estratificada: algunos resultados no eran añadidos a menos que otros resultados específicos tuvieran valores por encima de ciertos umbrales, ya que suponían requisitos más críticos para producir buenas armonizaciones.

Por otra parte, el primer autor en usar redes neuronales para composición algorítmica fue Todd [173] en 1989, quien usó una red neuronal recurrente de tres capas diseñada para producir una secuencia temporal de *outputs* que codificaban una melodía monofónica. *MUSACT* [19] utiliza redes neuronales para aprender un

## 2.1. Introducción y trabajo previo relacionado

---

modelo de armonía musical y fue diseñado para capturar las intuiciones musicales de cualidades armónicas. Por ejemplo, una de las cualidades de un acorde dominante es crear en el oyente la expectación de que un acorde de tónica está a punto de ser escuchado. Cuanto mayor es la expectación, mayor es la sensación de la consonancia del acorde de tónica. Los compositores pueden elegir satisfacer o violar estas expectativas en mayor o menor grado. *MUSACT* es capaz de aprender tales cualidades y generar expectativas graduadas en un contexto armónico dado. Otros intentos usando redes neuronales fueron los hechos en *NETGEG* [72] en 1996 y el modelo desarrollado por Coca et al. en 2011 [36].

Como se ha mencionado, la aplicación de algoritmos evolutivos a la composición algorítmica es proclive a usarse de forma híbrida junto con otras técnicas de inteligencia artificial. Una forma común de implementar una función de bondad es como una suma ponderada de las características de la composición (aunque el ajuste de los pesos para optimizar el algoritmo evolutivo puede resultar difícil, excepto para problemas muy simples). Por ejemplo, en el año 2000 Marques et al. [118], compusieron melodías polifónicas cortas utilizando una representación muy directa de los genotipos y una función de bondad simple, con una evaluación bastante simple de la armonía y el valor melódico. En 2004, Johnson et al. [96] también compusieron melodías cortas utilizando un algoritmo evolutivo con una función de bondad que es la suma ponderada de una serie de características muy básicas de la melodía. En 1998, Papadopoulos y Wiggins [134] implementaron un sistema que, dada una progresión de acordes, evolucionaba melodías de jazz utilizando como función de bondad una suma ponderada de ocho evaluaciones de características de la melodía, que van desde heurísticas muy simples sobre la velocidad y la posición de las notas hasta similitud entre fragmentos musicales especificados por el usuario. Un enfoque similar fue implementado por Harris [79] en 2008. Estos y otros trabajos, los cuales han usado evolución de composiciones musicales, han obtenido resultados que lejos de ser realmente significativos, son relativamente modestos.

A parte de los métodos ya mencionados, en la literatura también se pueden encontrar otras técnicas aplicadas a la composición algorítmica como es el caso de los autómatas celulares [87], las cadenas de Markov [181], [116], la inteligencia de enjambre [20] y la optimización por colonia de hormigas [69].

La mayoría de las estrategias para la composición por computadora se han centrado en la imitación de estilos humanos, sin embargo los nuevos enfoques de composición por computador deben intentar alejarse de esta tendencia, proveyendo al computador con conocimiento sobre composición musical (de igual forma que un aprendiz humano lo haría), de manera que pueda crear su propio estilo. *Melomics* sigue este enfoque desarrollando y aplicando con éxito un paradigma de desarrollo artificial a un dominio dentro de la creatividad computacional: los nuevos sistemas de composición algorítmica.

## **2.2. *Melomics*: un nuevo enfoque dentro del campo de la creatividad computacional**

*Melomics* tiene su origen en un proyecto de investigación, iniciado a finales de 2010, denominado *MELOMICS: Optimización de la respuesta terapéutica a la modulación de estímulos auditivos* [179], que fue desarrollado por la Universidad de Málaga, en colaboración con empresas TIC nacionales e internacionales. Este proyecto fue financiado por el Subprograma INNPACTO, perteneciente al Ministerio de Ciencia e Innovación, recibiendo una financiación de 1.3 millones de euros, que supuso hasta esa fecha la mayor cuantía recibida por la Universidad de Málaga por un proyecto de investigación. La redacción y gestión de la solicitud de financiación que permitió poner en marcha este proyecto supuso un riguroso y detallado trabajo, llevado a cabo por el autor de esta tesis en estrecha colaboración con su director y otros miembros de su grupo de investigación. Todo este proceso ha contribuido a la adquisición, por parte del autor, de capacidades de gestión investigadora, necesarias



## 2.2. Melomics: un nuevo enfoque dentro del campo de la creatividad computacional

para aspirar a obtener financiación económica que dé origen a futuros proyectos de investigación. Estas capacidades y habilidades, posibilitan dar el salto de la investigación básica en el ámbito universitario a la investigación aplicada que repercuta en los diferentes niveles del tejido productivo nacional e internacional. Las principales actividades y tareas llevadas a cabo durante la gestión de la solicitud de financiación que dio origen a este proyecto se enumeran de la siguiente manera:

- Análisis de requisitos legales y formales de la convocatoria de ayuda a la cual se presentó el proyecto.
- Redacción de memoria técnica y económica del proyecto.
- Contacto con empresas privadas y otras universidades para establecer consorcio y colaboraciones.
- Tramitación directa de gestiones con la O.T.R.I. (Oficina de Transferencia de resultados de Investigación).
- Administración de documentos, redacción de contratos y presupuestos.
- Tramitación de solicitud de financiación y posterior subsanación de documentación ante el Ministerio de Ciencia e Innovación.

Desde el punto de vista científico, el objetivo de este original proyecto de investigación es establecer un nuevo paradigma de música aplicada a la medicina usando técnicas de inteligencia artificial y “computación sentiente”. Las grandes posibilidades que brinda el núcleo de esta innovadora tecnología han permitido ampliar los horizontes iniciales de *Melomics* destacándose como un nuevo enfoque dentro del campo de la creatividad computacional. Más concretamente, se centra en la composición algorítmica, y tiene como objetivo la automatización completa del proceso de composición de la música profesional.

*Melomics* sigue un enfoque *evo-devo*, ya que combina un algoritmo evolutivo con un esquema de codificación indirecta para establecer correspondencia entre genotipos y fenotipos. En esencia, *Melomics* implementa la evolución de estructuras musicales

complejas, codificadas en genomas artificiales (se asemejan a los organismos vivos pluricelulares, que se desarrollan a partir de un genoma, y también evolucionan en el tiempo). Estos genomas representan la información musical de una manera indirecta y muy compacta: cada genoma codifica las especificaciones para generar una pieza musical siguiendo un complejo proceso de desarrollo. A medida que la población de composiciones evoluciona, las mutaciones realizadas en sus genomas provocarán cambios en las estructuras musicales resultantes. Estos cambios son evaluados por una función de bondad que determina si las nuevas estructuras están mejor o peor dotadas según ciertas restricciones musicales formales y principios estéticos básicos [50]. A medida que avanza la evolución, los genomas serán sometidos a transformaciones, haciendo a las piezas musicales correspondientes cada vez más complejas, y más adaptadas a los requerimientos.

### 2.2.1. Ordenadores compositores: *Iamus* y *Melomics109*

La tecnología de *Melomics* descrita se ha aplicado en dos sistemas informáticos: *Iamus* y *Melomics109*. *Iamus* (Figura 2.1a), por un lado, es un clúster de computación alojado en la Universidad de Málaga que ha producido decenas de partituras completas de música clásica contemporánea (Figura 2.2), siendo consideradas, por músicos profesionales y compositores, como indistinguibles de las escritas por compositores avanzados [17]. Por otro lado, *Melomics109* (Figura 2.1b), es un clúster de computación que se dedica completamente a la composición y la síntesis de música popular, un tipo de música diferente a la generada por *Iamus*, ya que está mucho más limitada por las restricciones culturales e instrumentales.

## 2.2. *Melomics*: un nuevo enfoque dentro del campo de la creatividad computacional



Figura 2.1: Sistemas informáticos que implementan la tecnología *Melomics*. (a) Clúster de computación *Iamus*. (b) Clúster de computación *Melomics109*.

Como se ha mencionado, *Iamus* es un computador-compositor especializado en la música clásica contemporánea. Lo que hace que su estrategia sea única es que, en lugar de aprender mediante ejemplos, se le enseña a componer de una forma similar a cómo los seres humanos aprenden las reglas de composición musical.



Figura 2.2: Ejemplos de partituras generadas por *Iamus*, consideradas por músicos profesionales lo suficientemente buenas como para haber sido escritas por un compositor avanzado [17].

El enfoque de desarrollo artificial implementado por ambos supercomputadores, proporciona maneras muy poderosas para la codificación de piezas musicales que abre el camino a una innovadora automatización de tareas creativas. Nuevas fusiones de géneros podrían ser descubiertos mediante operadores de recombinación que mezclen genomas musicales de diferentes estilos, por lo que los genomas descendientes pueden mostrar características combinadas de los genomas, acelerando la aparición de nuevos estilos musicales híbridos. También, los genes que codifican un *jingle* particular, podrían incorporarse en otro genoma, dando como resultado composiciones con fragmentos musicales que evocan la melodía original. Por último, un tema podría adaptarse mediante evolución a ciertas restricciones (basadas, por ejemplo, en un guión de video), adaptándose a las canciones que mejor se ajustan. Las aplicaciones de este tipo podrían ser disruptivas si se provee a la comunidad musical con herramientas eficaces para aplicarlas. Hacer esto no es nada fácil, y constituye uno de los principales problemas para la difusión de las soluciones basadas en inteligencia artificial.

En creatividad computacional, otro inconveniente es la naturaleza mal definida de los problemas bajo estudio. En el caso concreto de *Iamus*, una función de bondad determina las condiciones bajo las cuales los organismos musicales evolucionan. Esta función se ha diseñado en colaboración con un músico profesional, el Dr. G. Díaz-Jerez, y evalúa las composiciones musicales de acuerdo con diversos criterios básicos de naturaleza estética y formal, permitiendo a poblaciones de piezas musicales evolucionar hacia formas musicales formalmente correctas y potencialmente bellas. A modo de ilustración, los acordes de piano de seis notas (que se tocan con una sola mano) penalizarían la composición por un criterio ergonómico, ya que no se pueden ejecutar. Además, la polifonía de algunos instrumentos es recompensada, ya que combinan bien (los resultados son estéticamente válidos). En total, cerca de un millar de reglas han sido codificadas

## 2.2. *Melomics*: un nuevo enfoque dentro del campo de la creatividad computacional

---

para definir el entorno en el que la música contemporánea considerada como correcta debe evolucionar.

Pero es la naturaleza *evo-devo* del algoritmo que hay detrás de *Melomics* lo que constituye un hito en el diseño de los ordenadores compositores totalmente autónomos, ya que proporciona las herramientas para acceder, manipular y convertir en lenguaje musical un mundo extraordinariamente diverso de estructuras complejas. Como se describe en el artículo dedicado a *Iamus* por la prestigiosa revista *Nature*: “...*unfolding complex structure from a mutable core has enabled the kind of dramatic invention found in biological evolution*” [9]. Lo que es notable en la música compuesta por *Iamus* es que permite la definición de nuevas estéticas, no imitando el estilo que cualquier músico en particular ya haya desarrollado. Esto distingue a *Iamus* de la mayoría de los intentos anteriores dentro la composición algorítmica, convirtiéndose en una herramienta de gran utilidad para los compositores en cualquier estilo musical.

### 2.2.2. Del laboratorio al mercado: hitos y logros alcanzados por *Melomics*

Durante su desarrollo, *Melomics* ha alcanzando varios hitos importantes en el campo de la creatividad computacional dentro de la composición algorítmica. Por un lado, en julio de 2012, se publica *Iamus* (Figura 2.3), el primer álbum de estudio compuesto íntegramente por *Iamus* sin ningún tipo de intervención humana. El disco está compuesto por 10 temas (seleccionados de entre mil composiciones), que fueron grabados por la Orquesta Sinfónica de Londres y destacados solistas.

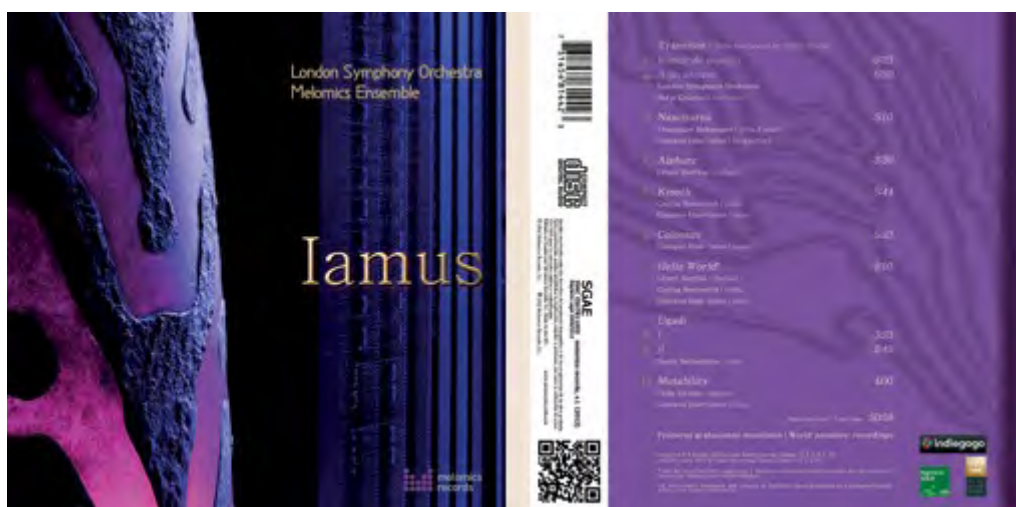


Figura 2.3: *Iamus*, primer disco compuesto íntegramente por el computador *Iamus* e interpretado por la Orquesta Sinfónica de Londres.

Por otro lado, si una solución basada en inteligencia artificial pretende ser realmente innovadora, requiere algo más que destreza en la ingeniería: tiene que interactuar de manera efectiva con el mundo real, solucionando problemas tecnológicos mediante la integración de estándares, haciendo un análisis preciso del mercado, y creando recursos a nivel hardware y software. Los investigadores y estudiantes de la UMA son animados a promover y financiar ideas innovadoras a través del concurso Spin-off de la Universidad, que se convoca desde 1998 para impulsar la transferencia de tecnología de los grupos de investigación hacia el tejido socio-económico. El objetivo de *Melomics* es la comercialización de esta generación de computadoras-compositores, siendo su propuesta galardonada en la XIII edición de este concurso.

*Melomics* ha considerado diferentes estrategias para hacer llegar esta tecnología al público en general. En principio, la generación de la música podría implementarse mediante la integración hardware de los algoritmos, o incrustarlos en un SoC (acrónimo del inglés, *System on Chip*). Estas soluciones requerirían un complejo control del espacio de parámetros por parte del usuario, unos tiempos de

## 2.2. *Melomics*: un nuevo enfoque dentro del campo de la creatividad computacional

---

composición y renderizado costosos, y también la distribución de caras librerías de música para la síntesis de audio. En su lugar, *Melomics* adoptó un enfoque diferente: usando el clúster *Melomics109*, ha creado y publicado en diciembre de 2013 un vasto repositorio de música popular (<http://www.melomics.com>) (Figura 2.4), cubriendo la mayoría de los géneros, formatos y necesidades, siendo navegable de una forma eficiente, utilizable por un conjunto de aplicaciones, y abierto a los desarrolladores a través de una *API* (acrónimo del inglés, *Application Programming Interface*).

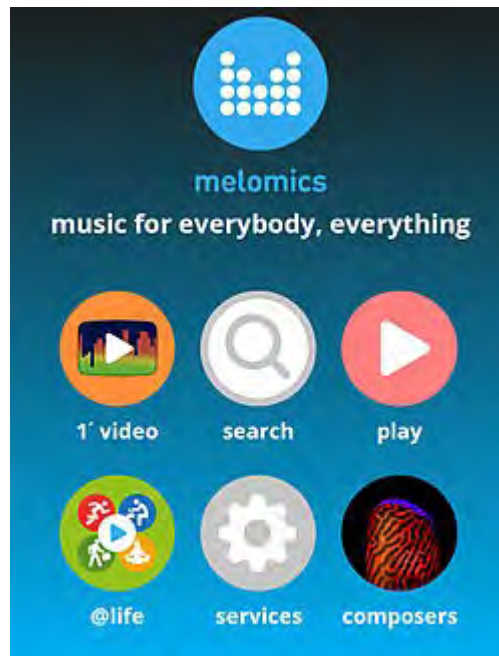


Figura 2.4: Página principal del repositorio creado por *Melomics109*, que cuenta con piezas de música popular abarcando todos los estilos esenciales. Estas piezas musicales son descargables bajo licencia CC0, de dominio público, sin restricciones de uso.

Este repositorio contiene piezas musicales descargables bajo licencia CC0, de dominio público y sin restricciones de uso. Estas piezas pueden ser directamente interpretadas por los músicos, o ser usadas como base musical por compositores. Para ello, las piezas musicales se presentan en una variedad de formatos, para cubrir diversos fines como edición musical (*MusicXML* y *MIDI*), interpretación humana

(partituras en *PDF*) y reproducción (calidad *MP3*). *MIDI* y *MusicXML*, formatos estándar para la representación de la música, pueden ser importados por la mayoría de los editores, lo que permite que sean libremente modificables por músicos profesionales. Las partituras en formato *PDF* permiten que la música esté lista para la interpretación instrumental. También se proporciona un archivo en formato *MP3* a través de un proceso personalizado de síntesis de audio, usando librerías de música profesionales para obtener resultados de alta calidad. Por último, la música se almacena y se pone a disposición del público general con una interfaz, la cual permite a los usuarios navegar de manera eficiente, seleccionando y adquiriendo temas.

El repositorio creado por *Melomics109* está allanando el camino para nuevos modelos de negocio, sirviendo de base para la creación de diversas aplicaciones móviles. Por ejemplo, *Melomics@life* (Figura 2.5a), una aplicación móvil que ofrece música por *streaming* para todos los escenarios de la vida cotidiana; *eMTCP* (Figura 2.5b), una aplicación de terapia musical para tratar las patologías de dolor crónico y trastornos del sueño; o *duermeteya* (Figura 2.5c), una aplicación destinada a ayudar a que los niños concilien el sueño. El hecho de que estas canciones estén libres de derechos de autor las pone a disposición de todo el mundo, lo que permite una amplia gama de opciones *freemium* (en contraste con las opciones que limitaban los márgenes de beneficios que permiten empresas como *Spotify* y *Pandora Media*). Por otro lado, también abre posibilidades para implementar “branding musical”, asociando melodías pegadizas a productos o marcas, y su inclusión (mediante la inserción de sus genes) en grandes piezas de música. Una *API* para los desarrolladores impulsará la realización de una clase de aplicaciones que no eran viables en la práctica hasta ahora.



## 2.2. *Melomics*: un nuevo enfoque dentro del campo de la creatividad computacional

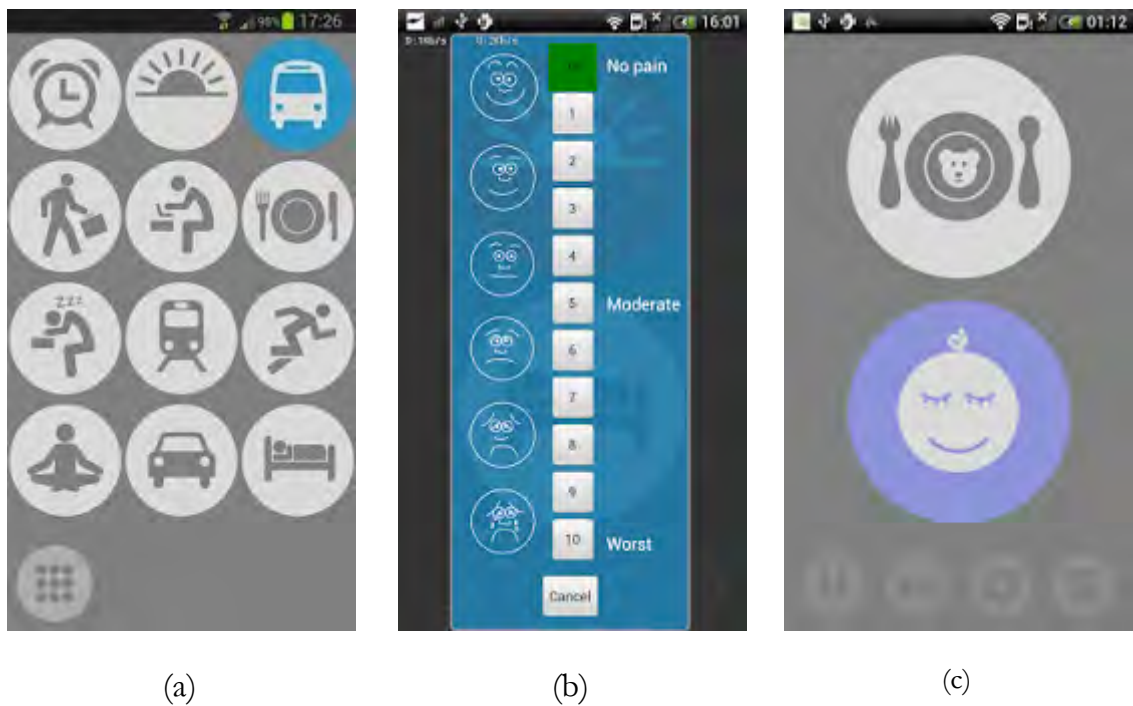


Figura 2.5: Aplicaciones móviles que ofrecen música por *streaming* usando la *API* de acceso a los contenidos del repositorio. (a) *Melomics@life*, adapta música a la actividad diaria, (b) *eMTCP*, proporciona terapia musical para tratar las patologías dolor crónico y trastornos del sueño y (c) *duermeteya*, ayuda a que los niños concilien el sueño.

Por último, otro hito reciente alcanzado por la tecnología *Melomics* ha sido la publicación en Julio de 2014 del álbum *Omusic* (Figura 2.6), primer álbum compuesto e interpretado por *Melomics109* sin ningún tipo de intervención humana. Este álbum contiene 12 temas que pertenecen a géneros populares y están disponibles en formato *MP3/MIDI* bajo licencia *CC0* (dominio público, libre y sin restricciones).



Figura 2.6: *0music*, primer disco compuesto e interpretado íntegramente por el computador *Melomics109*. Contiene 12 temas que pertenecen a géneros populares y están disponibles en formato *MP3/MIDI* bajo licencia *CC0*.

### 2.3. Validación de la tecnología *Melomics* como sistema de generación musical

La música compuesta por computador se está convirtiendo en un instrumento clave para medir las capacidades creativas de las computadoras y en algunos casos es potencialmente difícil distinguirla de la música compuesta por humanos. En este sentido, y con el fin de medir el grado de creatividad de la música generada por *Melomics*, se ha llevado a cabo un experimento para explorar en qué medida la música compuesta e interpretada por el supercomputador *Melomics109* tiene la misma valencia que la compuesta e interpretada por un humano. Salvando las distancias, este experimento puede considerarse como un test de *Turing* musical.

El objetivo de este estudio era doble. Por un lado se pretendía medir si los niveles de valencia, excitación y emoción generados por la música creada por *Melomics109* eran similares a la creada por un humano. Por otro, se cuestionaba a los participantes si eran capaces de determinar si una pieza musical había sido compuesta por *Melomics109* o por un humano.

Según Juslin et al. [97], la música tiene influencia en cómo se siente el oyente. De acuerdo a lo que hemos visto en este capítulo las computadoras son capaces de generar música desde la nada usando diferentes estrategias y algoritmos. El presente estudio constató que la gente siente emociones mientras está escuchando este tipo de música. Esta evidencia está en contradicción con la idea de que las computadoras no son capaces de generar emociones [167]. Sin embargo, la pregunta realmente interesante es si la gente siente emociones de forma equivalente cuando escucha música generada por computador y cuando ésta es creada por un ser humano. Sólo cuando la gente otorga la misma valoración a la música hecha por humanos y a la hecha por computador es posible determinar si las computadoras son capaces de crear música similar a la humana. Supper [168] sugiere que la composición algorítmica podría permitir la automatización de la creación de un número infinito de composiciones musicales. Por lo tanto las posibilidades de la música hecha por computador parecen no tener límites. Sin embargo, además de tener la capacidad de componer música, es necesario que esta música compuesta por computador sea capaz de evocar las mismas emociones que la música humana. De lo contrario, este tipo de música seguiría aún lejos de la música compuesta por humanos. Este estudio pretende comprobar si la música compuesta por *Melomics109* está al mismo nivel que la compuesta por los humanos.

#### 2.3.1. Participantes

En este estudio participaron de forma voluntaria un total de 251 sujetos, repartidos entre visitantes del Museo Interactivo de la Música de Málaga (MIMMA) y alumnos y profesorado del Conservatorio Superior de Música de Málaga. La edad de los sujetos estaba comprendida entre 20 y 60 años, siendo la edad media 34.2. En referencia al sexo, 124 de los participantes fueron hombres y 127 mujeres. La nacionalidad de los participantes fue muy diversa (22 países), con esta distribución:

- España: 79.6%
- Resto de Europa: 15.6 %
- América: 4.3 %
- Asia: 0.5 %

De los 251 participantes, 149 disponían de cinco o más años de formación académica musical, lo que garantizaba una muestra de sujetos con un alto grado de conocimiento en composición musical, confiriendo una mayor fiabilidad a los resultados obtenidos en el experimento.

### 2.3.2. Materiales

Se crearon dos piezas diferentes, una compuesta por un compositor humano, y otra por *Melomics109*, con las mismas especificaciones:

- estilo: balada de guitarra;
- instrumentos: piano, piano electrónico, bajo y guitarra eléctrica;
- compás: 4/4;
- tempo: 90 bpm;
- duración aproximada: 1:42 minutos;
- estructura: A B A´;
- escala: mayor.

Ambas piezas fueron interpretadas por un músico humano y también por *Melomics109* (sintetizadas automáticamente y mezclando la representación *MIDI* de la pieza hecha por músico humano, en el caso de *Melomics109*). Esto proporcionó las cuatro combinaciones siguientes (además de una pieza hecha de sonidos naturales<sup>1</sup>):

- CC (compuesta por computador, interpretada por computador)
- CH (compuesta por computador, interpretada por humano)

---

<sup>1</sup> [https://archive.org/details/Sounds\\_of\\_Nature\\_Collection:Jungle\\_River,Jungle\\_birdsong\\_and\\_Showers](https://archive.org/details/Sounds_of_Nature_Collection:Jungle_River,Jungle_birdsong_and_Showers), combinada con música de <http://www.freesfx.co.uk/>

### 2.3. Validación de la tecnología *Melomics* como sistema de generación musical

---

- HC (compuesta por humano, interpretada por computador)
- HH (compuesta por humano, interpretada por humano)
- NS (sonidos naturales).

Para recoger las respuestas de los participantes a las preguntas que se planteaban sobre las piezas musicales, se utilizó un cuestionario (en inglés y español) disponible tanto en formato papel como en formato interactivo a través de una tableta electrónica, además de unos auriculares aislantes que aseguraban la calidad de la audición de las piezas musicales durante el experimento. La Figura 2.7 muestra algunas imágenes del escenario y desarrollo del experimento. Para el diseño del experimento se contó con la colaboración y el asesoramiento del Dr. Alfredo Raglio, reconocido investigador en el campo de la terapia y comunicación musical.



Figura 2.7: Imágenes de la realización del experimento llevado a cabo. Para obtener las respuestas de los sujetos se utilizaron tanto medios electrónicos (tabletas) como cuestionarios en formato papel.

#### 2.3.3. Procedimiento

Antes de proceder a escuchar las piezas musicales, los participantes fueron cuestionados acerca de algunos datos tales como edad, sexo, país de origen y años de formación académica musical. A continuación, cada sujeto fue asignado aleatoriamente a un grupo (A o B) y escuchó cinco piezas musicales, de una

duración de 1:42 minutos cada una, en dos fases, tal y como se resume en la Tabla 2.1. El experimento se realizó a doble ciego: ni el investigador responsable del estudio, ni el participante eran conocedores del grupo asignado, y, por tanto, del orden en el que las piezas musicales eran presentadas.

Durante la Fase I, cada sujeto del grupo A (134 de 251) escuchó las tres piezas (HH / CH / NS) en orden aleatorio, para garantizar que el orden de escucha no influyese en las respuestas del sujeto. Como se ha indicado, HH y CH fueron compuestas por un ser humano y por *Melomics109*, respectivamente, y ambas interpretadas por un músico humano. Después de haber escuchado cada pieza, el sujeto debía responder a cinco cuestiones sobre si esa pieza se podría considerar música, las representaciones mentales que le generaba, y los estados emocionales que le evocaban. Concretamente las cuestiones eran las siguientes:

- (1) *¿Dirías que lo que has escuchado es música?*;
- (2) *¿Genera representaciones mentales al escucharlo?*;
- (3) *En caso afirmativo, ¿podrías escribir tres representaciones que hayan venido a tu mente mientras lo escuchabas?*;
- (4) *¿Genera estados emocionales?* ;
- (5) *En caso afirmativo, ¿podrías escribir tres sentimientos que hayas sentido mientras lo escuchabas?*.

	Grupo A	Grupo B
Fase 1	HH / CH / NS	CC / HC / NS
Fase 2	HH / CH	CC / HC

**Tabla 2.1:** Distribución de los participantes en dos grupos A y B. En el grupo A las piezas HH y CH son compuestas por humano y *Melomics109*, respectivamente. En el grupo B las piezas CC y HC son compuestas por *Melomics109* y humano, respectivamente. En ambos grupos, el intérprete varía (humano en grupo A y *Melomics109* en grupo B). Para ambos grupos el experimento se lleva a cabo en dos fases: Fase 1, en la que se cuestiona al participante sobre las emociones transmitidas por la pieza musical; y Fase 2, en la que se le pregunta si la pieza musical es compuesta por humano o computador.

En la Fase II, los participantes del grupo A escucharon de nuevo en orden aleatorio las piezas HH y CC, debiendo contestar a la siguiente pregunta: *¿Cree que esta pieza ha sido compuesta por un humano o por una computadora?* .

Los sujetos asignados al grupo B (117 de 251) procedieron de manera similar: escucharon las piezas musicales (CC / HC / NS) en orden aleatorio, CC y HC, compuestas por *Melomics109* y el compositor humano, respectivamente, pero en este caso ambas interpretadas por *Melomics109*. De esta manera, la interpretación no influía en los resultados del experimento, ya que para cada grupo (A y B) las piezas tienen diferente intérprete (humano para grupo A y *Melomics109* para grupo B).

#### 2.3.4. Resultados

Se realizaron comparaciones de tablas 2x2 mediante test de Ji-Cuadrado con corrección de continuidad, salvo en caso de frecuencias esperadas inferiores a 5, en el cual se empleó el test de Fisher, estableciéndose el nivel de significación en  $p < 0.05$ . Los resultados demuestran que los sujetos no fueron capaces de diferenciar entre la música compuesta por humano y la compuesta por *Melomics109*, ni siquiera sabiendo que una de las piezas era compuesta por un ordenador (Tabla 2.2). Concretamente en el grupo A, para la pieza HH (compuesta por humano) el 52% de los participantes contestó que era compuesta por computador y 48% que era compuesta por humano, mientras que para la pieza CH (compuesta por *Melomics109*), el 36% contestó que era compuesta por computador y el 64% que era compuesta por humano. Con respecto al grupo B, para la pieza HC (compuesta por humano) el 50% de los participantes contestó que era compuesta por computador y 50% que era compuesta por humano, mientras que para la pieza CC (compuesta por *Melomics109*), el 53% contestó que era compuesta por computador y el 47% que era compuesta por humano. En cuanto a la medida de la generación de emociones y la evocación de representaciones mentales, los resultados reflejaron que la música compuesta por *Melomics109* se encuentra prácticamente al mismo nivel que la música

compuesta por un humano. Sorprendentemente, según los resultados, los sonidos naturales fueron tan buenos como la música suscitando emociones, e incluso mejores evocando representaciones mentales.

		Compuesta por humano	Compuesta por <i>Melomics109</i>
Grupo A	HH	52%	48%
	CH	64%	36%
Grupo B	HC	50%	50%
	CC	47%	53%

Tabla 2.2: Resultados del experimento realizado. Los sujetos no fueron capaces de diferenciar entre la música compuesta por humano y la compuesta por *Melomics109*, ni siquiera sabiendo que una de las piezas era compuesta por un ordenador.

## 2.4. Validación del uso potencial de *Melomics* como sistema de tratamiento terapéutico

Como se ha mencionado en la sección 2.2, la tecnología desarrollada por *Melomics* tiene su origen en un proyecto de investigación que tiene por objetivo establecer un nuevo paradigma de la música aplicada a la medicina usando técnicas de inteligencia artificial y “computación sentiente”. Con este propósito, *Melomics Music Medicine* ( $M^3$ ), es un subsistema de *Melomics* que proporciona al paciente una música adaptada utilizando un teléfono inteligente u otro dispositivo, existiendo también la posibilidad de captar el estado fisiológico del paciente y ofrecer música que afecte positivamente a ese estado en tiempo real. Este sistema puede ser utilizado para prevenir y tratar varias condiciones tales como la ansiedad, el estrés, el dolor crónico y trastornos del sueño. El uso de esta tecnología podría mejorar la eficacia del tratamiento y reducir el uso de fármacos con la consecuente disminución de los efectos secundarios para los pacientes.



Con el objetivo de validar el uso potencial de la música generada por *Melomics* en entornos clínicos, se ha realizado un estudio para evaluar y validar el efecto del sistema terapéutico *M<sup>3</sup>* en la reducción de la percepción del dolor sobre una población pediátrica seleccionada al azar que se somete a la prueba de punción cutánea (*SPT*, acrónimo del inglés *Skin Prick Test*).

### 2.4.1. La prueba de punción cutánea (*SPT*) y el uso de la música en el ámbito hospitalario

En la atención ambulatoria pediátrica, uno de los métodos más utilizados de forma rutinaria en el diagnóstico de la alergia es la prueba de punción cutánea (*SPT*). En este procedimiento se colocan varias gotas de los alérgenos en la superficie de la piel del antebrazo con una separación mínima de dos centímetros aproximadamente, para evitar falsos-positivos, y se introducen en la epidermis con una aguja normalizada. Este tipo de prueba es prácticamente indolora ya que no existe sangrado debido a que la aguja apenas penetra en la piel. Sin embargo, en ocasiones puede producirse bastante picazón cuando se producen resultados positivos. A pesar de su carácter indoloro, muchos niños lloran durante esta prueba, sintiendo miedo y dolor, lo que podría representar una experiencia traumática para este tipo de pacientes. Hasta la fecha no hay muchos estudios clínicos que establezcan el grado de dolor de la prueba de punción cutánea.

Por otra parte, la música es una herramienta ampliamente usada para tratar diferentes enfermedades tradicionalmente relacionadas con los trastornos neurológicos y psiquiátricos. Sin embargo, el uso de la música en el ámbito hospitalario ha ampliado su campo de aplicación en los últimos años. En estos casos, la música reduce la ansiedad y el dolor derivado de procedimientos quirúrgicos o médicos, antes y durante las intervenciones [192], [120], [108]. La música también se ha utilizado para reducir el dolor en ausencia de procedimientos

médicos, por ejemplo, en la condición de dolor crónico [82], [76], [14], [27]. Una reciente revisión sobre la música y el dolor [30], llegó a la conclusión de que escuchar música puede reducir el nivel de intensidad del dolor percibido, así como el requerimiento de opiáceos en tipos concretos de dolor y poblaciones. Además, se han llevado a cabo varios estudios que investigan el dolor y la ansiedad en el ámbito pediátrico [80], [191]. A pesar de los diferentes modelos y técnicas utilizados en la práctica de la terapia musical (ya sea psicológica o neurocientífica, receptiva o activa), la escucha directa de música parece ser el enfoque más adecuado para tratar la condición de dolor como una alternativa a la medicación. En estos casos, la música suele ser elegida por los pacientes, teniendo en cuenta el hecho de que la preferencia personal es más eficaz que el uso de música propuesta por un terapeuta. La música preferida de un paciente puede activar algunos sistemas neuroquímicos asociados a la motivación y el placer, pero también puede estar unido al estrés y la excitación, modulando algunos sistemas tales como los de dopamina, el cortisol y la serotonina. [32]. En otros casos, la música es adaptada por el terapeuta siguiendo criterios específicos para la relajación o activación de los pacientes, en relación con el contexto y el objetivo terapéutico. A pesar de los efectos positivos de la música en la disminución del dolor (ampliamente reportado en la literatura), no existen herramientas para que las personas puedan controlar con facilidad y eficacia su dolor de una forma habitual y cómoda.

### 2.4.2. Participantes

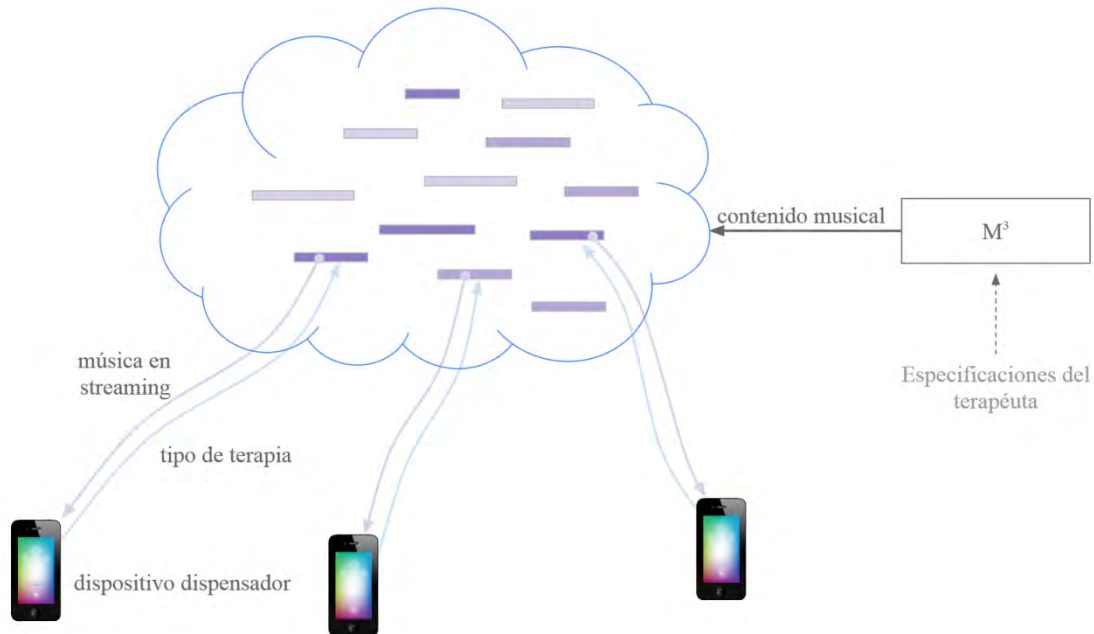
Un total de 72 niños fueron incluidos en este estudio y se sometieron a un procedimiento *SPT* en el Hospital Infantil de Málaga. Los niños fueron asignados al azar a un grupo experimental (36) y a un grupo de control (36). Cuatro niños en el grupo experimental fueron descartados debido a reacciones vagas (2), la manipulación del dispositivo durante el procedimiento (1) y probable deficiencia

mental (1). Por lo tanto, el grupo experimental finalmente incluyó a 32 niños. En cuanto a la distribución por sexo y edad, el 54.8% de los niños del grupo experimental eran varones frente al 59.5% en el grupo control. La edad media en el grupo experimental era de 9 años, y 7.5 años en el grupo de control.

### 2.4.3. Materiales

*M<sup>3</sup>* ha sido diseñado para ser replicable y transferible a cualquier unidad médica de una manera directa. Se implementa un enfoque de música en *streaming* (Figura 2.8), donde la música está almacenada en la nube, y libremente accesible a través de un navegador web (<http://melomics.com/SPT>), o una aplicación para Android (<http://melomics.com/SPTapp>), por lo que los pacientes pueden usar este servicio con cualquier dispositivo que pueda conectarse a Internet (ya sea un ordenador, una tableta o un teléfono inteligente).

Para este estudio, la música de *Melomics* fue generada de acuerdo con dos principios fundamentales que aparecen en la literatura: (1) la posibilidad de adaptar las piezas teniendo en cuenta las características de la música para niños; y (2) la necesidad de crear una música de distracción (parametrizando el proceso de composición de una manera que hace que sea atractiva y con variaciones), para desviar la atención del dolor y la ansiedad debido al procedimiento *SPT*. Al igual que en el experimento anterior, se contó con la colaboración del Dr. Alfredo Raglio para el diseño de la música y de los diferentes aspectos relacionados con el experimento, así como con la inestimable colaboración de parte del personal médico del área de pediatría del Hospital Infantil de Málaga.



**Figura 2.8:** Descripción general del sistema terapéutico  $M^3$ . Un dispositivo de suministro con conexión a Internet (ya sea un teléfono inteligente o cualquier ordenador convencional) está configurado para un tipo concreto de tratamiento. Este dispositivo solicita la música a un servidor en la nube o repositorio, que ofrece música en un estilo concreto (tal como se especifica por un terapeuta, y generada por *Melomics*). El dispositivo también incluye un reproductor, que proporciona la música al paciente, generalmente a través de un auricular.

Este enfoque musical se suscribe dentro del campo de la medicina musical [81] en el que las características de los estímulos musicales juegan un papel central en la tarea terapéutica, independientemente de la relación entre el paciente y el terapeuta, que caracteriza a otros enfoques de musicoterapia. Con el objetivo de evaluar en qué medida puede la música de *Melomics* reducir la percepción del dolor durante el procedimiento de *SPT*, se utilizaron principios basados en la evidencia de la práctica de la musicoterapia [1], [60], [59], [180].

#### 2.4.4. Procedimiento

Durante el procedimiento *SPT*, los niños pertenecientes al grupo experimental estuvieron escuchando música generada por *Melomics*, mientras que los niños del grupo de control no recibieron ningún estímulo auditivo especial. La música fue dispensada por medio de un teléfono inteligente en el que se instaló *M<sup>3</sup>*, y auriculares. La música fue diseñada para ser atractiva y alegre para los niños, con el fin de evitar que su atención se centrara en el procedimiento al que eran sometidos. La intensidad del dolor experimentado por los niños durante el procedimiento se evaluó a través de una Escala Visual Analógica (EVA) 0-10 (0: sin dolor, 10: dolor severo) [61], (Figura 2.9)



Figura 2.9: Escala Visual Analógica (EVA) 0-10 (0: sin dolor, 10: dolor severo)

La muestra fue descrita usando medidas de dispersión estadística (mediana y rango intercuartil) para las variables cuantitativas, mientras que se utilizó la distribución de frecuencias para variables cualitativas. Con el fin de comparar el grupo experimental con el grupo control, se utilizó la prueba de Chi-cuadrado para analizar las variables cualitativas independientes, mientras que se utilizó la prueba de Mann-Whitney para las cuantitativas independientes. Las variables resultado, presencia de dolor (moderada o intensa), Reducción Relativa del Riesgo (RRR) y número necesario a tratar (NNT) se calcularon con intervalos de confianza del 95% (IC95%), respectivamente, estableciendo un nivel de significación estadística  $p < 0.05$ . Además, se realizó un análisis cuantitativo y cualitativo de los datos usando el programa estadístico SPSS.

El consentimiento informado se obtuvo de los padres de todos los niños y el protocolo de investigación fue aprobado por el Comité Ético del Hospital Infantil de Málaga.

### 2.4.5. Resultados

Con respecto a la intensidad del dolor, de acuerdo con la medida cuantitativa EVA, la mediana fue de 2 para los niños del grupo experimental, en comparación con una mediana de 4, para los niños del grupo de control. La diferencia entre los dos grupos es estadísticamente significativa ( $p < 0.001$ ). Estos resultados se muestran en la Figura 2.10.

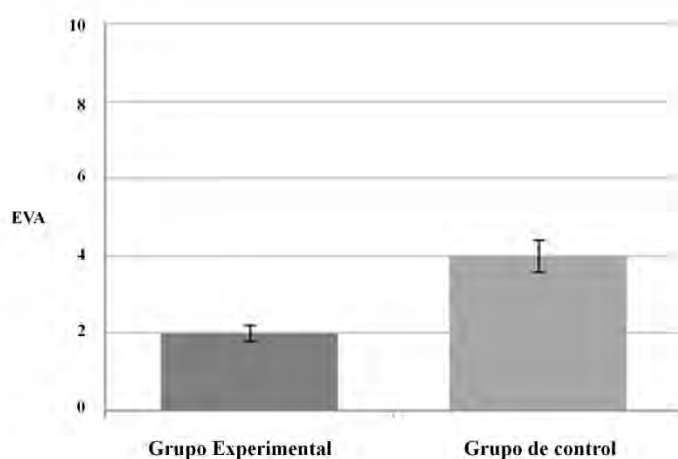


Figura 2.10: Valor de la mediana de la EVA en el grupo experimental y el grupo control ( $p < 0.001$ ).

Otro resultado interesante es que la presencia de dolor (EVA 3 a 10) se reportó en un 29% de los niños del grupo experimental, en comparación con 81.1% en el grupo control ( $p < 0.001$ ). Tomando en consideración esta medida cualitativa, se obtuvo una RRR del 64% (IC95%: 39-79) y un NNT de 2 (IC 95%: 1-3). Estos resultados se muestran en la Figura 2.11.

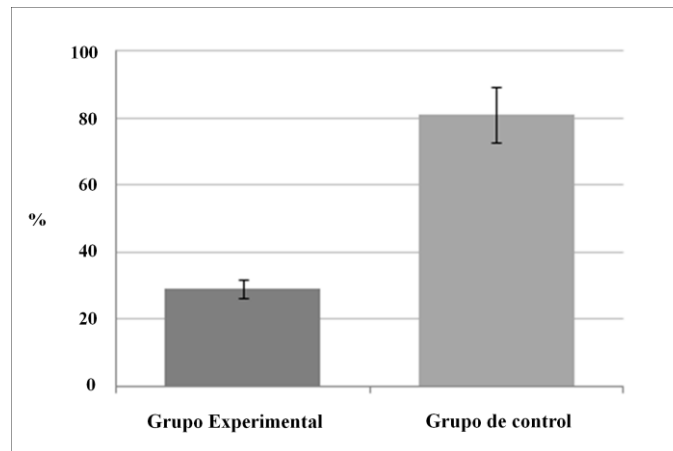


Figura 2.11: Porcentaje de niños que reportaron presencia de dolor (valores entre 3 y 10 en la EVA) en los grupos experimentales y de control ( $p < 0.001$ ).

## 2.5. Conclusiones

La inteligencia artificial puede beneficiarse mucho de su aplicación en otros campos, y *Melomics* y los sistemas derivados de esta tecnología no son más que un ejemplo. La inteligencia artificial actualmente lidera el uso de modelos bioinspirados, y seguirá generando interesantes aplicaciones basadas en sistemas *evo-devo*. Los resultados presentados en este capítulo han logrado un importante hito en la creatividad computacional conectando dos dominios diferentes: la computación evolutiva con desarrollo artificial y el lenguaje musical. La información sobre *Iamus* ha sido ampliamente cubierta por los medios de comunicación<sup>2</sup>, como el primer equipo que compone obras completas de música clásica contemporánea profesional usando su propio estilo. Diez de estas obras han sido grabadas y compiladas en un álbum de debut, el primero de todos que ha sido totalmente compuesto por una inteligencia no humana por sí misma, e interpretada por músicos de primera categoría (incluyendo la Orquesta Sinfónica de Londres).

---

<sup>2</sup> <http://melomics.com/press>

Sin embargo, hoy en día no es suficiente con que una computadora sea capaz de crear música más o menos agradable, sino que lo realmente interesante desde el punto de vista de la creatividad computacional, es si esta música puede transmitir las mismas emociones y evocar las mismas representaciones mentales que la música creada por humanos. Con el fin de medir el grado en el que la música generada por la tecnología *Melomics* tiene la misma valencia que la compuesta e interpretada por un humano, se han presentado los resultados de un experimento controlado llevado a cabo en el Museo Interactivo de la Música de Málaga (MIMMA) y el Conservatorio Superior de Música de Málaga. Estos resultados vienen a determinar que la música generada por *Melomics* es prácticamente indiferenciable de la música hecha por un humano, tanto desde el punto de vista emocional como evocador, siendo los participantes incapaces de distinguir las piezas compuestas por humano y las piezas compuestas usando la tecnología *Melomics*.

En cuanto a la aplicación industrial de este enfoque *evo-devo* a la composición musical, se prevé tres áreas principales de aplicación. Primero, proporciona novedosas técnicas para la industria de la música: el repositorio de música generado por *Melomics109* que permite a personas sin formación musical producir música original. Como consecuencia directa, esto también abre nuevas oportunidades de negocio: un mercado en el que la música se comercializa como una materia prima; y estrategias publicitarias para el “branding musical”. Segundo, también ofrece nuevas herramientas para la manipulación de la música, al igual que la obtención de variaciones de un tema (por mutación del material genético), así como la fusión automática de géneros (por recombinación de temas de diversos géneros). Tercero, introduce el concepto de “computación sentiente” que lee información de los sensores de los dispositivos móviles para ofrecer música adaptada a las necesidades del usuario, abriendo la posibilidad a la creación de aplicaciones en diferentes campos, entre ellos el de los sistemas terapéuticos.



## 2.5. Conclusiones

---

Conectando con esta tercera vía de aplicación, se diseñó un experimento para determinar el efecto de  $M^3$ , un subsistema derivado de *Melomics*, como sistema terapéutico en la reducción de la percepción del dolor, en comparación con el protocolo estándar sin música, en un grupo de población pediátrica seleccionada al azar durante un procedimiento de *SPT*. Los resultados mostraron efectos estadísticamente significativos en la reducción de la percepción del dolor. Sólo el 29% de los niños que utilizaron  $M^3$  reportó un dolor moderado o intenso, en comparación con el 81% de los niños que no escucharon música. Esto significa que el riesgo de sufrir dolor es 2.8 veces mayor con el protocolo estándar. Una posible explicación de estos resultados significativos podría ser la confirmación de la hipótesis de que la música parametrizada para tal fin desvía la atención de los niños del dolor, y, en consecuencia, reduce la percepción del dolor. También podría ser que el soporte tecnológico (la presencia de un teléfono inteligente), y los estímulos visuales podrían influir en los resultados del estudio. Sin embargo, lo interesante de estos resultados es la posibilidad de crear un catálogo de música a medida y un sistema de distribución estándar, que tiene en cuenta (1) las características de la música preferida por los niños, y los parámetros de sonido específicos, dirigidos a captar la atención de los niños, y (2) las necesidades de una instalación médica donde se realiza el procedimiento. Los resultados del estudio deben interpretarse con precaución, debido al pequeño tamaño de la muestra y la ausencia de otro grupo de control para comparar los efectos de la música tradicional (por ejemplo, la música comercial dirigida a niños) y la música generada por *Melomics*. Por otra parte, este estudio supone una prueba controlada aleatoria que demuestra una fuerte eficacia en la reducción de la percepción del dolor utilizando un método estandarizado, fácil de implementar y económico. Además, este mismo enfoque puede ser utilizado en diferentes entornos clínicos, también en otras condiciones, como para reducción de la ansiedad antes de intervenciones quirúrgicas, alivio de trastornos del sueño, estrés, dolor crónico, etc. Otra ventaja es que  $M^3$  es una herramienta terapéutica que los pacientes pueden utilizar de forma gratuita con

simplemente disponer de una conexión a Internet. A pesar de los resultados prometedores planteados por el estudio realizado, se necesitan más estudios basados en la evidencia para evaluar las potencialidades terapéuticas de la música de *Melomics*, y su aplicación en el ámbito clínico.

## Capítulo 3

# Evolución de avatares animados mediante física como soporte creativo para la sincronización de personajes y música en escenas de animación

En este capítulo se propone una herramienta de creatividad computacional, para la sincronización automática de personajes y música en animaciones coreográficas. El personaje o avatar es modelado como un conjunto de barras (*struts*) y resortes elásticos (*springs*); y simulado usando el motor de física *ODE* (acrónimo del inglés, *Open Dynamics Engine*). Los resortes tienen la capacidad de responder, expandiéndose o contrayéndose, al ritmo de una pieza musical de entrada el cual es determinado mediante un algoritmo de análisis de señal. La dinámica global de estos resortes reactivos, genera una animación del avatar dirigida por la música, imitando de una manera abstracta el complejo proceso creativo y de expresividad emocional que emerge en un ser humano, al bailar al son de una pieza musical. La simulación física del modelo permite alcanzar cierto grado de realismo en los movimientos y la automatización de los mismos, evitando el coste temporal que supondría componer la animación usando otras técnicas tales como la captura de movimientos o el *keyframing*. Parámetros como la sensibilidad con la que los resortes captan la música o los grados de libertad de las extremidades del avatar son codificados de forma directa mediante un genoma. Una población de estos genomas es evolucionada usando una función de bondad interactiva mediante el cual, el usuario determina su grado de satisfacción con la coreografía ejecutada por el personaje para un determinado tema musical de entrada. Con la finalidad de aplicar este modelo en la práctica, se propone una herramienta mediante la cual la animación obtenida para un avatar es transformada de forma automática en un formato adecuado para ser

importada por el software de animación *Blender* (aunque la animación de la armadura generada podría ser exportada a otros paquetes software de animación). De esta forma el modelo propuesto se convierte en una herramienta de creatividad computacional que puede ser usada de forma directa o como semilla generadora de animaciones coreográficas, por animadores amateurs o profesionales, para la creación de escenas de animación. Esta herramienta pretende ahorrar tiempo y esfuerzo situándose como una solución intermedia entre las técnicas actuales basadas en captura de movimientos y el *keyframing* en cuanto a relación realismo-coste temporal.

El capítulo está organizado de la siguiente manera. La sección 3.1 presenta una introducción a los fundamentos de la animación basada en física y la sincronización de personajes y música, así como una revisión de los trabajos previos relacionados. La sección 3.2 describe el algoritmo de detección de ritmo musical utilizado, basado en el análisis de señal. El modelo es descrito en la sección 3.3 así como las características del algoritmo evolutivo utilizado. En la sección 3.4 se presenta la herramienta de creatividad computacional basada en el modelo descrito que sirve de soporte a los animadores en el proceso creativo. Por último, las conclusiones derivadas de este modelo son discutidas en la sección 3.5.

### **3.1. Introducción y trabajo previo relacionado.**

El modelo presentado en este capítulo se alinea dentro del campo de la animación de personajes dirigida por música, enriquecido con las características de realidad en los movimientos y su automatización que aportan las técnicas de animación basada en física. Por este motivo la introducción y revisión de trabajos previos se presenta en dos partes: centrada en primer lugar en el campo de la animación basada en física y posteriormente en el contexto de la sincronización de personajes y música, en escenas de animación.

#### 3.1.1. Animación basada en física

La animación basada en física ofrece un enfoque innovador dentro de la animación por computador. En lugar de la técnica tradicional denominada *keyframing* que manipula directamente las trayectorias de los movimientos de objetos y personajes, este enfoque permite que todo movimiento sea el resultado de un proceso de simulación de la física. Como consecuencia, los personajes y los objetos basados en física interactúan de forma automática de una manera que es físicamente muy aproximada a la realidad, sin la necesidad de datos o secuencias de comandos de movimiento adicionales. La idea de utilizar la simulación física para animar personajes virtuales fue reconocida por primera vez por Armstrong [5] y Wilhelms [187]. El tipo de personajes que se usan en animación basada en física son mayoritariamente modelos humanos, sin embargo algunos trabajos usan modelos animales [141], [73], [44], [170], robots [86], [141], o criaturas ficticias [161], [170]. Estos modelos a menudo se simplifican para aumentar el rendimiento de la simulación y para facilitar su control. Los personajes animados por física son más difíciles de controlar que los generados por métodos basados en *keyframing*, ya que son controlados indirectamente mediante fuerzas y torsiones internas o externas de los elementos que los componen. El resultado de esta reducción en el control es la apariencia robótica de algunos modelos [132], [183], [106], lo que se conoce con el concepto de valle inquietante (del inglés *uncanny valley*) [157]. Otro problema al que se enfrentan los modelos de animación basada en física es su coste computacional, estando algunos trabajos recientes [48], [93] limitados a un solo personaje en simulaciones en tiempo real. No obstante, pese a estos problemas, la animación basada en física está recibiendo un gran interés en los últimos años.

Dentro del campo de la animación basada en física existen varios enfoques que utilizan la física para animar objetos y personajes. La simulación de cuerpos rígidos (*Rigid body simulation*) se caracteriza por el uso de objetos no deformables con propiedades físicas tales como masa o densidad sobre los que pueden aplicar

primitivas como translación o rotación [83]. Por su parte los objetos y personajes propios de la simulación de cuerpos blandos (*soft body simulation*), a diferencia de los hechos de cuerpos rígidos, pueden deformar su cuerpo. Trabajos previos en el área de la simulación de cuerpos blandos han demostrado que sistemas de mallas de muelles se pueden utilizar para simular el movimiento de gusanos, serpientes y peces [124], [178], [74]. La simulación de fluidos suele ser algo más costosa desde el punto de vista computacional, utilizándose métodos de *Lagrange* o *Euler* para su simulación [67]. Los sistemas de partículas son una técnica ampliamente utilizada para la creación de efectos visuales en películas y videojuegos debido a su sencilla implementación. El ciclo de un sistema de partículas está compuesto por tres fases: generación, simulación y extinción. El modelo propuesto por Sims [162], hace uso de la computación paralela para la animación y renderización de sistemas de partículas. Por último, la técnica de *flocking* modela el comportamiento complejo de enjambres de agentes tales como pájaros, peces o insectos mediante fuerzas virtuales. El programa de simulación desarrollado por Reynolds, *Boids* [143] supuso el trabajo pionero en la aplicación de esta técnica, del cual se han derivado otros trabajos propuestos recientemente [49], [84].

Por último, en cuanto a la implementación de los modelos, destacar que los diferentes trabajos en el campo de la animación basada en física utilizan varios motores de simulación física tales como *ODE* (Open Dynamics Engine) [164], Bullet [45], PhysX, Newton, o Havok, siendo *ODE* el más utilizado.

### 3.1.2. Sincronización de personajes y música en escenas de animación

El proceso de sincronización de una animación con una pieza musical sigue suponiendo hoy en día una ardua tarea para los profesionales de la animación. El objetivo de la mayoría de los métodos de sincronización es tomar una animación ya existente y sincronizarla para que los cambios de movimiento se alineen con el ritmo de una determinada pieza musical. Los gráficos de transición de movimiento son

### 3.1. Introducción y trabajo previo relacionado.

---

una técnica muy popular para lograrlo. Los enfoques propuestos por y Kim et al. [98] y Alankus et al. [2], utilizan gráficos de transición para sintetizar nuevas secuencias de movimiento a partir de datos de captura de movimiento. Shiratori et al. [159], [158], también dividen los datos de captura de movimiento originales en secciones más pequeñas para sintetizar el nuevo movimiento. En este método, la similitud rítmica entre segmentos musicales y de movimiento se utiliza para sincronizar la animación con la pieza musical. Algunos autores utilizan curvas de movimiento en lugar de los datos de movimiento, como el caso de Cardle et al. [29], que propusieron un sistema que genera el movimiento editando directamente los fotogramas clave en la señal de movimiento. Existen técnicas que apoyan la construcción de movimientos complejos a partir de movimientos más simples. En este sentido, los trabajos de Fod et al. [65] y Woch y Plamondon [190], defienden que el movimiento complejo se puede simplificar en una combinación de movimientos básicos llamados primitivas. Las secuencias de baile de gran duración pueden ser divididas en rutinas que consisten en movimientos de baile separados. Los movimientos de baile individuales son las primitivas que se combinan entre sí para crear rutinas de baile.

Por otro lado existen trabajos que se basan en un enfoque opuesto al anterior, utilizando la pieza musical como dato de entrada para dirigir y guiar la animación del personaje. Este enfoque, denominado animación de personajes dirigida por música, elimina la necesidad de tener que sincronizar los movimientos de una animación ya existente con la música de entrada, ya que se utiliza esta música como fuente de origen para generar la animación. Dentro de este enfoque se encuentran los trabajos de Sauer y Yang [153], que proponían un sistema de *scripts* para definir los movimientos básicos de un estilo de baile, los cuales posteriormente eran adaptados a la música de entrada, componiendo la animación. El modelo de Dubbin y Stanley [57], hacía evolucionar el baile de personajes que eran controlados por redes neuronales artificiales (ANNs), analizando la música de entrada con una transformada rápida de Fourier (FFT). Un modelo similar, que analizaba el ritmo

musical, pero aplicado al baile de robots fue el propuesto por Guangyu et al. [75]. Hahn y Diefenbach [78] aplicaron esta filosofía para animar la dinámica de videojuegos de plataformas que era guiada por la música de la escena. Como se ha mencionado, el modelo propuesto en este capítulo se encuadra dentro del enfoque de la animación de personajes dirigida por música.

### 3.2. Algoritmo de detección de ritmo musical

La función del algoritmo de análisis musical implementado es analizar, en tiempo real, una señal de entrada contenida en un archivo en formato *WAV* o *MP3*, y extraer de ella ciertas características musicales, concretamente el patrón de *beats* que determina de una forma sencilla el ritmo musical. Esta secuencia de *beats* resulta uno de los aspectos más reconocibles de un tema musical [154], sobre todo para personas sin formación musical. Sentir los *beats* de un tema musical resulta un proceso natural, mediante el cual un ser humano es capaz de moverse o simplemente golpear la superficie de una mesa con su mano al ritmo de la música. Como se muestra más adelante, la posición de estos *beats* será usada como dato de entrada que guiará el movimiento del personaje. Para la implementación del algoritmo se han utilizado las librerías *PyAudio* y *Wave* del lenguaje *Python*.

Las señales de audio analizadas por el algoritmo tienen una tasa de frecuencia de muestreo de 44.1 KHz, que equivale 44100 *samples* por segundo. Esta señal de entrada puede representarse computacionalmente como un vector unidimensional de *samples*,  $v$ , cuyos valores de amplitud determinan una cierta energía. Un *beat* supondrá una variación drástica de esta energía en un determinado instante, en comparación con la energía media local dentro una ventana o buffer de *samples*, por lo que obtener el patrón de *beats* se reduce a detectar estos picos de energía. Dos parámetros importantes del algoritmo son el tamaño de *samples* utilizado para detectar la energía instantánea y el tamaño de la ventana o buffer usado para calcular



### 3.2. Algoritmo de detección de ritmo musical

---

la energía media local (Figura 3.1). Las pruebas realizadas nos han llevado a determinar que los valores óptimos para estos parámetros son 1024 y 44032 *samples*, respectivamente. Formalmente el cálculo de la energía instantánea viene determinado por la fórmula:

$$Ei = \sum_{k=i_0}^{i_0+1024} v[k]^2$$

Mientras que el cómputo de la energía media en un buffer de 44032 *samples* de una señal con una tasa de 44100 *samples* por segundo viene determinada por:

$$Em = \frac{1024}{44100} \times \sum_{k=0}^{k+44032} v[k]^2$$

El funcionamiento general del algoritmo se describe a continuación. Al inicio del procesamiento, se tendrán en cuenta los valores de los primeros 1024 *samples* de  $v$  para calcular la energía instantánea, y se escogerá una ventana o buffer formado por los 42032 primeros *samples*, cuyos valores determinarán la energía media local para ese buffer. Un *beat* es detectado en los 1024 *samples* si la energía instantánea es superior a la energía media local, es decir si se cumple la condición:

$$Ei > Em \times S$$

, donde  $S$  es una constante que determina la sensibilidad del algoritmo en la detección de *beats*, y que debe ser ajustado dependiendo del género musical que se quiere analizar. Por ejemplo, en géneros musicales como el rap o el tecno, los *beats* suelen ser claros y precisos por lo que un valor suficiente de  $S$  sería  $S \geq 1.5$ ; sin embargo, en géneros como el pop o el rock donde existe más ruido en la señal, los *beats* suelen ser algo más difusos por lo que habría que aumentar la sensibilidad del algoritmo utilizando un valor de  $S$  alrededor de 1.1. Continuando con el funcionamiento del algoritmo, tras analizar los primeros 1024 *samples* se escogen los

siguientes 1024 para calcular la energía instantánea y se mueve 1024 *samples* a la derecha la ventana para calcular la energía media, determinando si existe un *beat* en este caso, y así sucesivamente hasta llegar al final de  $v$ . Este algoritmo ha sido probado para diferentes tipos de géneros musicales ajustando el valor de  $S$ , obteniéndose resultados bastante satisfactorios en la mayoría de los casos. Como se ha comentado, la señal de entrada puede ser un archivo en formato *WAV* o *MP3* (convertido automáticamente por el algoritmo a formato *WAV*). Lo interesante es que la señal es suministrada por *streaming*, por lo que la detección de *beats* se realiza en tiempo real paralelamente a la escucha del tema musical o sonido de entrada.

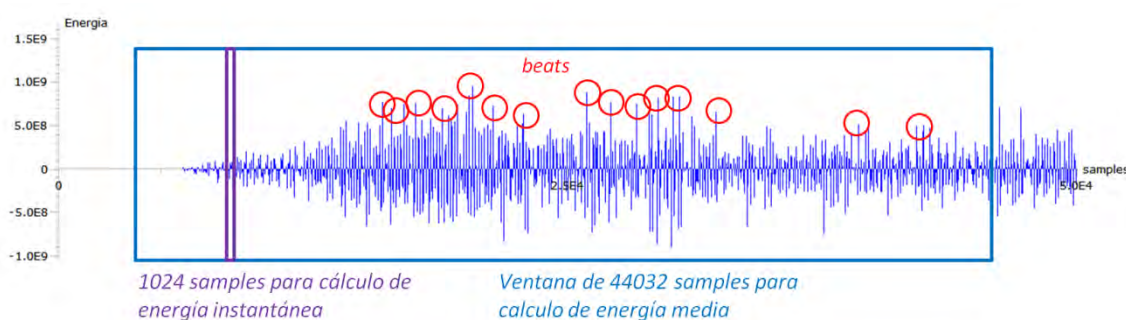


Figura 3.1: Algoritmo de detección de *beats* para una señal de entrada. La energía instantánea  $E_i$  es calculada mediante la suma cuadrática del valor de 1024 *samples* (color morado). Esta energía es comparada con la energía media  $E_m$  de una ventana o buffer formada por 44032 *samples* (color azul), de forma que si la energía instantánea es mayor que la energía media multiplicada por un parámetro de sensibilidad  $S$ , es decir  $E_i > E_m \times S$ , significa que se ha detectado un *beat* en ese instante. En rojo se puede apreciar un ejemplo de algunos de los *beats* que se detectarían en la señal de entrada mostrada.

### 3.3. Descripción del modelo

En esta sección se describe un modelo evolutivo con función de bondad interactiva para la búsqueda de animaciones coreográficas en avatares animados mediante física. En una primera instancia nos centraremos en explicar cómo se modelan los avatares, para posteriormente introducir la física del modelo. Por último se

### 3.3. Descripción del modelo

---

describen las características del algoritmo evolutivo utilizado para llevar a cabo la búsqueda evolutiva.

#### 3.3.1. Modelado del avatar

Se ha modelado un avatar con forma de personaje humano utilizando el motor de física *ODE*, a través de la implementación de esta librería disponible para el lenguaje de programación *Python*. El avatar está compuesto por barras y esferas definidas por unas propiedades físicas tales como longitud, radio, peso y densidad. Estos elementos permanecen unidos mediante articulaciones que respetan, en cierta medida, los grados de libertad de las articulaciones del cuerpo humano. Las restricciones de los grados de libertad de la unión entre dos elementos vienen determinadas por el tipo de unión y el ángulo de giro sobre los ejes de coordenadas  $x$ ,  $y$ ,  $z$  con respecto a la posición inicial de reposo de ambos elementos. Por ejemplo, en el caso de la articulación del codo se utiliza una unión o *joint* de tipo bisagra o *hinge* con un ángulo de libertad de  $120^\circ$  alrededor del eje  $x$ , imitando el movimiento de brazo y antebrazo unidos por el codo. La animación del avatar es dirigida por varios resortes elásticos que conectan diferentes puntos de su cuerpo tales como mano izquierda–mano derecha y cadera derecha–hombro derecho. En la Figura 3.2 se puede observar la representación de un avatar en el modelo propuesto. A partir de una entrada de datos especificada por una tema musical en formato *WAV* o *MP3* reproducido en *streaming*, se utiliza el algoritmo de detección de ritmo musical para detectar los *beats*, de forma que los resortes reaccionarán en tiempo real ante la presencia de un *beat*, contrayéndose o expandiéndose durante un instante con una determinada intensidad  $I \in [0.1, 0.9]$  específica para cada resorte. Para un determinado tipo de música, puede ser interesante que el movimiento de los brazos sea más enérgico que el de las piernas, por lo que la intensidad con la que los resortes de los brazos reaccionan a los *beats* será mayor. Este modo de funcionamiento puede ser interpretado como una capacidad de los resortes para

“sentir” la música, lo que denominamos resortes reactivos o “sentientes”. Como resultado, la dinámica global que emerge en el conjunto de resortes, se traduce en un movimiento animado del avatar. El algoritmo de detección de ritmo musical asegura en todo momento la sincronización entre los movimientos del avatar y la música de entrada. Como se puede apreciar en la Figura 3.2 el color de los resortes indica su estado de tensión, siendo rojo cuando el resorte se encuentra en compresión, azul en expansión y verde en reposo. El color negro se utiliza para determinar que un resorte no está activo por lo que no interviene en la dinámica de la animación. La dinámica física de estos resortes es explicada con más detalle en la siguiente sección.

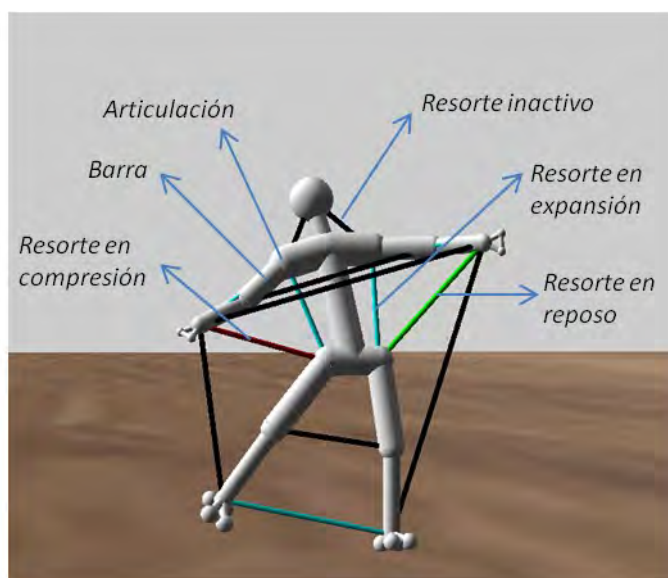


Figura 3.2: Representación de un avatar simulado usando el motor de física *ODE*. El avatar está compuesto por barras y esferas unidos mediante articulaciones cuyos grados de libertad respetan en cierta medida las propias del cuerpo humano. Varios resortes elásticos unen diferentes puntos del avatar. La variación de las dinámicas de estos resortes en función de su capacidad para “sentir” la música de entrada determinan el movimiento global del avatar. El color de los resortes indica su estado de tensión, siendo rojo en el caso de estar en compresión, azul en el caso de estar en expansión y verde cuando se encuentra en estado de reposo. Los resortes de color negro son resortes inactivos (en la práctica no existen) por lo que no influyen en la dinámica del avatar.

La utilización de resortes elásticos ha sido ampliamente utilizada para simular las características biomecánicas de los cuerpos de organismos biológicos en muchos

### 3.3. Descripción del modelo

---

modelos [124], [113], [95], ya que simulan las propiedades elásticas y de amortiguación de elementos tales como músculos, tendones o ligamentos. En este modelo, su utilización difiere ligeramente ya que estos resortes no se corresponden con ningún elemento del cuerpo, sino que lo que se pretende es aprovechar las mencionadas propiedades de los resortes como medio para captar y representar la expresividad emocional del cuerpo humano al escuchar música. Un ser humano que baila al ritmo de un tema musical puede ser considerado como una especie de marioneta cuyos movimientos son guiados y alimentados por el ritmo musical. Además, las propiedades de amortiguación de los resortes contribuyen a que los movimientos sean suaves, ofreciendo un comportamiento natural del avatar.

#### 3.3.2. Física del modelo

El avatar es simulado en un entorno o escenario virtual físicamente realista, lo que permite que a su vez la animación generada tenga un alto grado de realismo, al mismo tiempo que se reduce el control necesario sobre el movimiento del avatar, ya que en un alto grado es guiado por las dinámicas físicas del sistema. Como se ha comentado, se ha utilizado el motor de física *ODE* tanto para la construcción del avatar como para el entorno virtual. La dinámica de los resortes viene determinada por la constante del resorte, la constante de amortiguamiento, y la longitud en reposo, siendo los dos primeros globales para todos los resortes, y el último particular para cada uno de ellos. Estos valores configuran las propiedades de compresión y tracción del resorte. El hecho de que un resorte se estire o se comprima en un momento dado durante la simulación depende de los parámetros y las fuerzas aplicadas sobre el mismo de forma interna (en el caso de contracción o expansión ante un *beat*); y de su interacción con el entorno y resto de elementos del avatar. Todos los resortes del avatar comparten las mismas constantes de resorte y amortiguación ( $k$  y  $c$ , respectivamente), mientras que la longitud de reposo difiere para cada uno y es definida en la posición inicial del avatar (posición de descanso).

La dinámica del resorte resulta de aplicar estas ecuaciones sobre los puntos que une:  $\vec{F}_k = -k\vec{l}$  (ley de Hook), y  $\vec{F}_c = -c\vec{l}$  (fuerza de amortiguación), donde  $\vec{l}$  es el vector de desplazamiento del resorte (es decir, la distancia y la dirección en la que se deforma el resorte). Durante la simulación, la dinámica de los resortes se ve afectada en cada paso de iteración bien sea por las fuerzas internas o externas.

Las barras y esferas que componen el cuerpo del avatar se ven afectadas por la fricción con el suelo. Esto se implementa de acuerdo con la expresión  $\vec{F}_m = -\mu_m \vec{v}$ , donde  $\mu_m$  es el coeficiente de fricción, y  $\vec{v}$  es el vector de velocidad de la barra o esfera en contacto con el suelo. Además de estas fuerzas, la simulación también incluye la fuerza gravitatoria  $\vec{F}_g = -g\vec{u}_z$ , donde  $g$  es la aceleración constante de la gravedad, y  $\vec{u}_z$  es el vector ortonormal al eje Z. Con el fin de reducir el coste computacional de las simulaciones se ha desactivado la detección de colisiones entre las barras y esferas del avatar, no así entre los elementos del avatar y el suelo del escenario. Los resortes carecen de presencia física ya que representan fuerzas de resorte y amortiguación, por lo que no existen colisiones con el resto de elementos.

### 3.3.3. Evolución de avatares mediante función de bondad interactiva

La exploración del espacio de búsqueda de coreografías del avatar descrito para un tema musical de entrada, resulta desmesuradamente amplio y es impracticable y poco eficiente acometerlo de forma manual. Con el objetivo de realizar esta tarea de optimización, se ha integrado el modelo en un algoritmo genético, una variedad de los algoritmos evolutivos. Los algoritmos evolutivos constituyen una de las principales técnicas propuestas en el campo de la inteligencia artificial para generar y explorar productos y resultados creativos [42].

Se sometieron a evolución poblaciones de avatares animados mediante un mismo tema musical de entrada. El genotipo de cada individuo se representa como una fila

### 3.3. Descripción del modelo

---

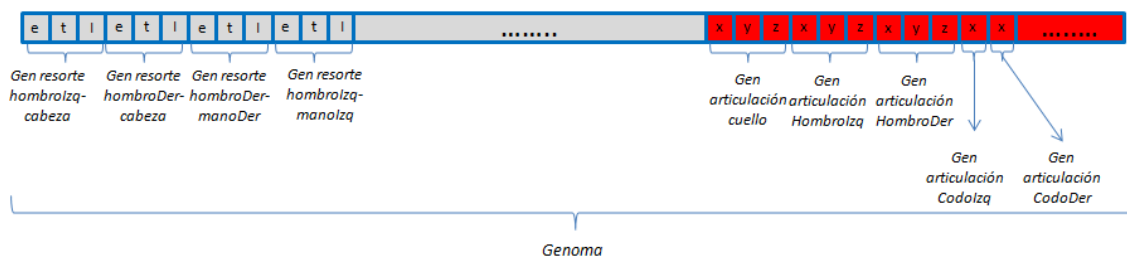
o vector que codifica la información de las propiedades de los resortes y las restricciones para los ángulos de los grados de libertad de las articulaciones. La información para cada resorte o articulación se considerará un gen dentro del genotipo.

El gen de cada uno de los catorce resortes que conectan el avatar se compone de tres parámetros: (1) *estado* ( $e$ ), que indica si el resorte está activo o inactivo y que será representado por los valores 1 y 0, respectivamente; (2) *tipo* ( $t$ ), que determina si el resorte reacciona ante un *beat* comprimiéndose o expandiéndose, representándose con los valores 0 y 1, respectivamente y (3) *intensidad* ( $I$ )  $\in [0.1, 0.9]$  que especifica la fuerza con la que el resorte se contrae o se expande ante un *beat*. El primer parámetro del gen es el más importante, ya que en el caso de tener un valor igual a 0 significa que el resorte se encuentra inactivo y no interviene en la dinámica general del movimiento del avatar (en la práctica es como si no existiese), por lo que el valor de los parámetros *tipo* e *intensidad* carecerían de interés para ese gen.

Por otro lado, para los genes de las once articulaciones correspondientes a hombros, codos, rodillas, cuello, tobillos y caderas, se codificarán parámetros (desde uno a tres) que determinan el ángulo de giro en radianes sobre los ejes de coordenadas  $x$ ,  $y$ ,  $z$  con respecto a la posición inicial de reposo. Los valores de estos parámetros estarán comprendidos en el intervalo  $[lmin, lmax]$ , siendo  $lmin$  y  $lmax$  el valor de los ángulos en radianes que determinan la amplitud mínima y máxima respectivamente para cada articulación del cuerpo humano, en cada uno de sus grados de libertad. Por ejemplo, para el caso de la articulación codo, cuyo grado de libertad se produce solo en un eje (concretamente en el eje  $x$  de nuestro modelo) se utilizará sólo un parámetro. Por otro lado, el límite máximo  $lmax$  para el ángulo del grado de libertad de esta articulación sería  $120^\circ$  ( $2\pi/3$ ) que corresponde con doblar el codo intentando subir la mano al hombro, entendiendo que la posición de reposo se establece con el brazo completo extendido apuntando al suelo en dirección vertical.

El valor de  $l_{min}$  para esta articulación es  $0^\circ$  ya que no se puede doblar el codo hacia atrás. Por el contrario la articulación cadera (unión pierna-cintura) en el cuerpo humano presenta movimiento en los tres ejes  $x, y, z$ , por lo que se representará con tres parámetros. Para el caso del eje  $x$ ,  $l_{max}$  tendrá un valor de  $120^\circ$  ( $2\pi/3$ ) que corresponde con levantar la pierna hacia arriba, y  $l_{min}$  tendrá un valor de  $45^\circ$  ( $\pi/4$ ) correspondiente a lanzar la pierna hacia atrás. Determinados valores de los grados de libertad de estas articulaciones permiten adaptar el movimiento del avatar a un determinado tipo de baile. Por ejemplo, en el caso del ballet, puede resultar interesante que el avatar levante una pierna a cierta altura mientras que la otra ofrezca menos movimiento, imitando en cierta manera el movimiento de baile típico de una bailarina de ballet.

La disposición espacial de los veinticinco genes que conforman el genoma se establece de forma secuencial como se muestra en la Figura 3.3. Los primeros catorce genes del genoma (situados a la izquierda) corresponden a los genes que codifican las propiedades de los resortes, mientras que los once últimos genes (situados a la derecha) determinan el valor de los grados de libertad de las articulaciones.



**Figura 3.3: Representación del genotipo de un individuo.** Se puede apreciar la existencia de genes que codifican los parámetros de los resortes (color gris) y genes que codifican los ángulos de los grados de libertad de las articulaciones (color rojo). Existen articulaciones como la que representa el codo que sólo usan un parámetro que indica el ángulo de giro expresado en radianes sobre el eje  $x$ , mientras que otras articulaciones, como la del cuello, codifican tres parámetros ya que rota en los tres ejes de coordenadas  $x, y, z$ .



### 3.3. Descripción del modelo

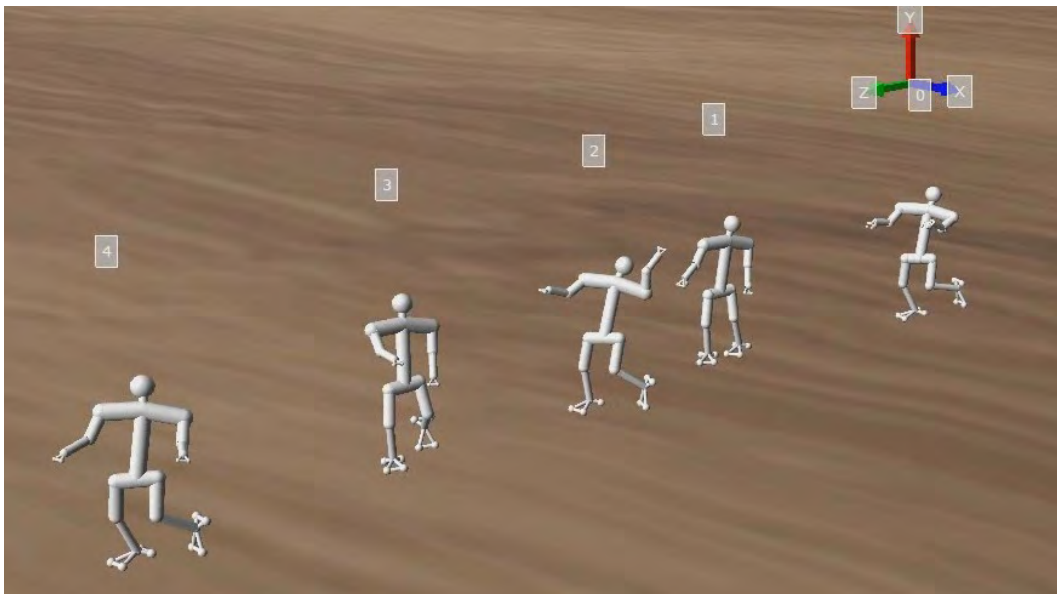
---

De una forma similar al modelo propuesto por Dubbin y Stanley [57], la evaluación del valor de bondad de cada individuo se realiza interactivamente por el usuario de acuerdo a los principios de la Computación Evolutiva Interactiva (*CEI*). La *CEI* es un campo dentro del aprendizaje automático que se sirve de la capacidad de los humanos para emitir juicios subjetivos [169]. Esta técnica ha sido utilizada en la evaluación de poblaciones en muchos sistemas evolutivos tales como las criaturas virtuales de Sims [161]. En este enfoque, se presentan al usuario varias alternativas visuales y éste debe evaluar su bondad. Esta información es utilizada para crear la nueva generación, repitiéndose el proceso hasta que al usuario se le presenta una solución que según su criterio satisface el problema. La falta de automatización de este sistema presenta el inconveniente de que el usuario puede llegar a cansarse al no encontrar una solución adecuada [169]. Sin embargo, en nuestro caso, creemos que la evolución de avatares animados al ritmo de la música puede resultar lo suficientemente interesante como para evitar que el usuario llegue a cansarse durante el proceso evolutivo.

Se sometió a evolución a poblaciones de un tamaño fijo de cinco individuos cuyos genes se inicializaron con valores aleatorios. El pequeño tamaño de la población se debe al hecho de la utilización de una función de bondad interactiva, ya que se ha considerado que un número mayor que cinco es demasiado elevado para que el usuario pueda fijarse con detalle en el comportamiento de cada individuo. En cada iteración del algoritmo, y haciendo uso de una interfaz sencilla, el usuario asigna un valor de bondad de entre 0 a 10 a cada individuo (Figura 3.4). Cuatro, de los cinco individuos que forman la nueva generación se obtienen mediante la selección por ruleta de dos individuos de forma directamente proporcional a su valor de bondad, y posterior cruce en un punto de los mismos para dar lugar a dos individuos más. A estos cuatro individuos se les aplica una mutación con una probabilidad de 0.4 (un valor relativamente alto con el objetivo de obtener diversidad en la población), consistente en la selección aleatoria de un parámetro de un gen y su modificación al

azar. El quinto individuo se obtiene mediante elitismo de forma que el que obtiene mayor puntuación siempre pasa a la siguiente generación.

Para las pruebas evolutivas se utilizaron cuatro piezas musicales correspondientes a géneros tan variados como la música clásica, el pop, rock o la electrónica. Para cada uno de estos géneros, el valor de la constante  $S$  que determina la sensibilidad del algoritmo de detección de *beats* fue de 1.01, 1.2, 1.25 y 1.3, respectivamente, con el objetivo de lograr una precisión adecuada en la detección del ritmo musical. La media de generaciones del algoritmo necesarias para obtener un comportamiento de baile satisfactorio fue de 49, siendo el tiempo medio invertido aproximadamente de 12:20 minutos, debido a que en la mayoría de las ocasiones el usuario sólo necesita 20 o 25 segundos para emitir valoraciones para los avatares de una generación.



**Figura 3.4: Asignación interactiva del valor de bondad de los individuos.** En cada iteración del algoritmo los cinco individuos de la población son simulados en un escenario. El usuario, mediante una sencilla interfaz, asigna una puntuación o valor de bondad entre 0 y 10 a cada individuo. Esta puntuación determina el grado de satisfacción del usuario con respecto al comportamiento del avatar. La búsqueda evolutiva termina cuando el comportamiento de uno o varios de los avatares cumple con las expectativas del usuario. Con el objetivo de eliminar elementos distractores los resortes elásticos permanecen invisibles durante la simulación.

### 3.4. Herramienta de soporte creativo para animación de personajes.

La descripción del modelo presentado hasta ahora en este capítulo, permite evolucionar una población de avatares con la finalidad de explorar el espacio de búsqueda de coreografías para un determinado tema musical de entrada. Sin embargo, su utilidad práctica sólo se reduce a obtener una grabación en vídeo de la simulación generada. En este sentido, con el objetivo de conferir al modelo una aplicación práctica y de utilidad tecnológica se ha desarrollado un sistema basado en *scripts* para exportar automáticamente el comportamiento del avatar evolucionado usando el motor de física *ODE*, a un formato que puede ser importado por el software de animación *Blender*. Crear una animación de un personaje utilizando suites de animación de este tipo es una tarea casi inabordable para usuarios con poca experiencia. A este aspecto se une la compleja y amplia interfaz característica de estas aplicaciones software. Si además, el usuario tiene que crear una animación del personaje sincronizada con música, esto se convierte en un trabajo prohibitivo o que requiere un coste de tiempo desmesurado. En esta sección se describe una herramienta que sirve de soporte para las tareas creativas tanto de usuarios amateurs como profesionales, simplificando en gran medida el proceso.

La Figura 3.5 muestra una visión general de la conexión entre el modelo propuesto para evolucionar avatares animados por música y la herramienta desarrollada para exportar los comportamientos obtenidos a una suite software de animación, como *Blender*. Una vez que mediante evolución se ha obtenido un avatar que satisface al usuario, se genera para este avatar un fichero de texto, *cuerpo.txt*, que especifica la estructura corporal del avatar en el estado inicial de reposo, concretamente las coordenadas  $x$ ,  $y$ ,  $z$  de las articulaciones que unen todas las piezas que componen su cuerpo. De forma paralela la herramienta también proporciona otro fichero, *animacion.txt*, que contiene la posición de las articulaciones para cada paso de tiempo de la simulación física obtenida en *ODE*. Esta información equivale a la generación

automática de los *keyframes* de la animación del avatar describiendo espacialmente todos movimientos del mismo para cada instante de tiempo.

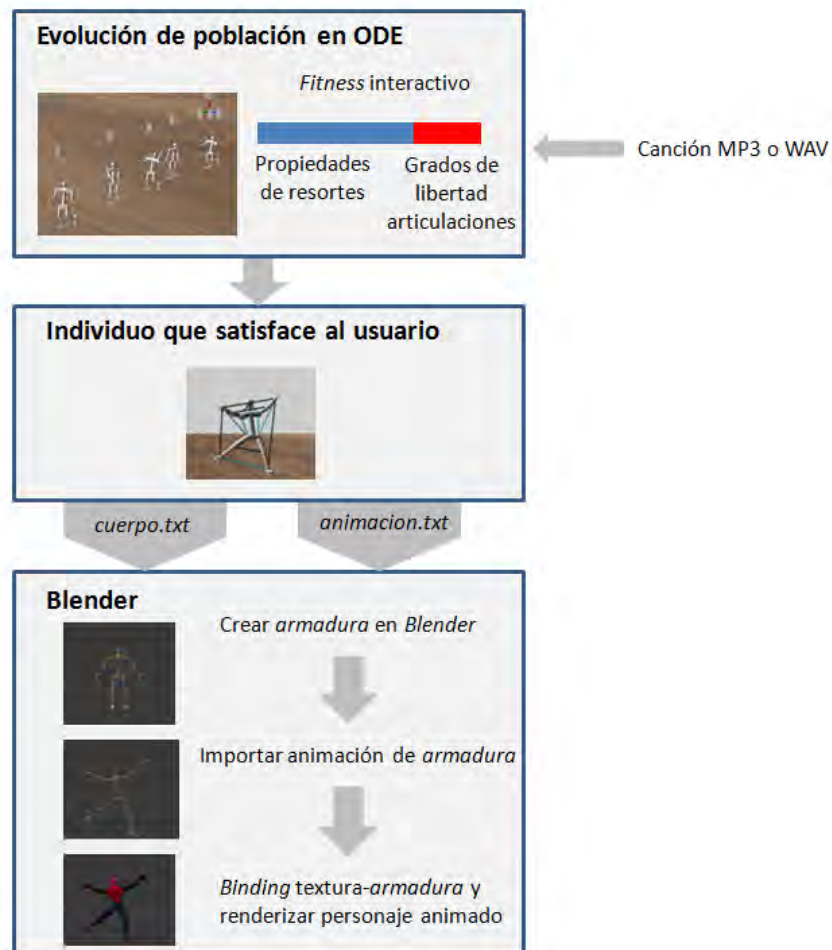


Figura 3.5: Esquema del modelo y herramienta propuestos para la animación automática de coreografías. En la simulación en *ODE* se evoluciona un conjunto de avatares mediante función de bondad interactiva. Para el avatar resultante de la búsqueda evolutiva, se generan dos ficheros, *cuerpo.txt* y *animación.txt*, que contienen la información espacial del cuerpo y de la animación, respectivamente. Esta información es importada en *Blender* para obtener automáticamente una armadura animada a la cual se le aplicará una textura para la posterior renderización del personaje animado.

La importación de esta información se lleva a cabo una vez dentro del software de animación *Blender*. Mediante un *script* se crea un objeto *armadura* a partir del fichero *cuerpo.txt* que equivale al esqueleto del personaje. Posteriormente, otro *script* carga

### 3.4. Herramienta de soporte creativo para animación de personajes.

---

los datos de la animación contenidos en el fichero *animación.txt* para dotar al objeto *armadura* de la animación. Como resultado se obtiene una armadura animada en *Blender*, que mapea la coreografía generada por búsqueda evolutiva en *ODE* de una forma automática y con poco esfuerzo para el usuario. En este punto del proceso el usuario puede utilizar directamente la animación, o puede modificar libremente ciertos movimientos a través de la ventana de edición de *keyframes* disponible en la interfaz de *Blender*. De una forma u otra, el último paso para terminar completamente un personaje listo para ser incluido en una escena de animación consistirá en dotar a la armadura de una textura mediante un proceso de *binding*, tarea que ya depende de las habilidades del animador. La Figura 3.6 muestra una secuencia de baile de un personaje en su versión en *ODE*, armadura y renderizado de textura en *Blender*, respectivamente. Se puede apreciar como los movimientos de la animación se exportan con detalle desde la simulación del avatar en *ODE* al personaje final renderizado en *Blender*. La Figura 3.7 muestra un ejemplo de una secuencia de baile de la *armadura* de un personaje evolucionado para bailar ballet y su posterior renderizado e inclusión en una escena.

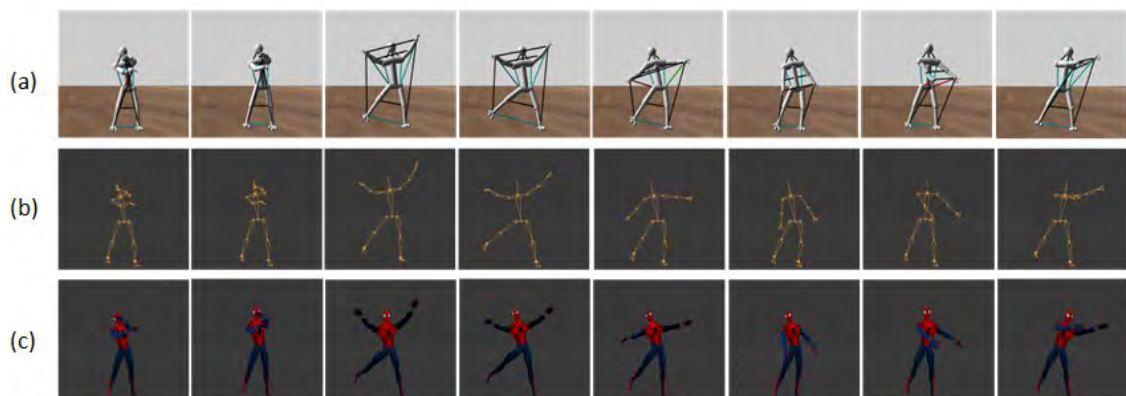
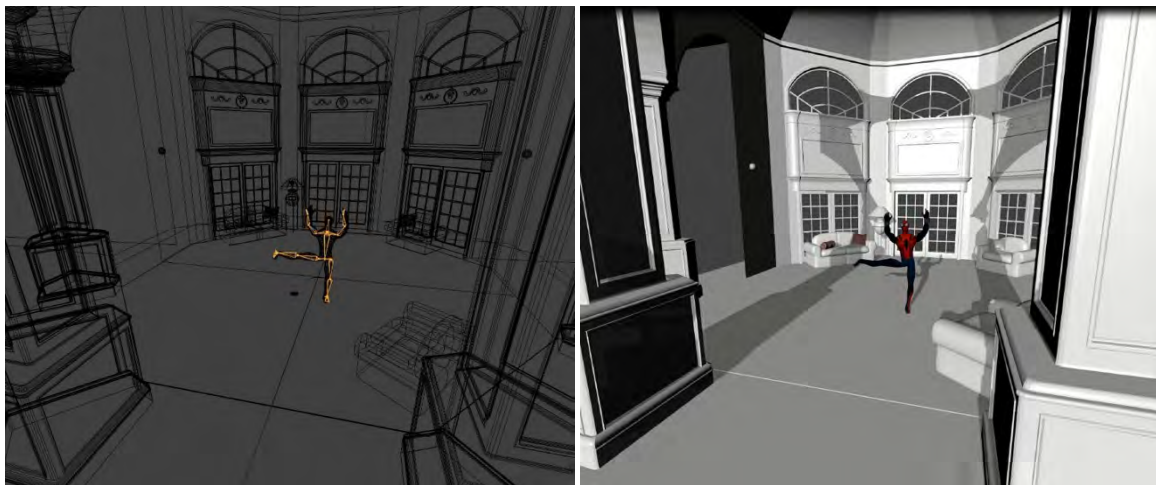


Figura 3.6: Visualización del avatar en tres estados del proceso de creación de la animación. (a) Simulación en ODE para su evolución (b) Animación de armadura (c) Aplicación de textura y posterior renderizado.



(a)

(b)

**Figura 3.7:** Inclusión de un personaje evolucionado para bailar ballet en una escena. (a) Armadura animada del personaje sin textura (b) Aplicación de textura y renderizado de escena.

### 3.5. Conclusiones

En este capítulo se ha presentado un modelo evolutivo con codificación directa para la animación de personajes dirigida por música y su integración en una herramienta para la generación automática de animaciones. La sincronización de personajes con música en escenas de animación coreográfica sigue siendo hoy en día un arduo trabajo que requiere una gran cantidad de tiempo y habilidad por parte de los animadores. Haciendo uso de la computación evolutiva y su capacidad de explorar el espacio de búsqueda de soluciones esta herramienta pretende servir de soporte creativo tanto para animadores aficionados como profesionales. El objetivo es proporcionar automáticamente a estos usuarios soluciones que pueden aplicar directamente, o utilizarlas como ideas desde las que partir a la hora de abordar el trabajo creativo que supone la creación de una animación coreográfica, sincronizada adecuadamente con un determinado tema o pieza musical. La novedad de la utilización de resortes elásticos con capacidad para “sentir” la música, permite a las animaciones generadas con este modelo adquirir el realismo y atractivo visual

### 3.5. Conclusiones

---

característico de las aproximaciones derivadas del campo de la animación de personajes basada en física.

El enfoque propuesto también supone una abstracción en cierta medida del conjunto de mecanismos que permiten la emergencia de la expresividad emocional de los seres humanos al escuchar música, planteando la cuestión de si el ser humano, al bailar, se comporta de manera reactiva, moviéndose por impulsos originados por picos de energía de la música que escucha. Como se ha descrito, estos picos de energía o *beats* que determinan el ritmo musical se han analizado y extraído mediante un simple algoritmo de análisis de señal.

Desde el punto de vista de la creatividad computacional este enfoque aprovecha y se beneficia de la ausencia de sesgo que caracterizan a las computadoras. Un animador humano que afronta la tarea creativa de la animación de un personaje sincronizado con música, tiende inconscientemente a escoger una serie de movimientos del personaje, los cuales probablemente se encuentran influenciados en mayor o menor medida por ideas preconcebidas de origen cultural, o por conocimientos previos. Por el contrario, una computadora que no es programada con ningún tipo de restricciones, aprovechando las capacidades de los algoritmos evolutivos, puede sistemáticamente explorar y obtener soluciones (en este caso concreto que nos ocupa comportamientos y movimientos de baile) que un animador humano quizás no podría concebir o plantearse por los motivos anteriormente mencionados.

De acuerdo a la relación realismo-coste temporal, las animaciones obtenidas con este modelo pueden situarse en un punto intermedio entre los enfoques de animación basada en captura de movimientos y la creación de una animación paso a paso usando *keyframing*. Por un lado, usando el modelo propuesto, el coste temporal de crear una secuencia de baile es drásticamente más bajo en comparación con el proceso casi artesanal que supone el uso de *keyframing*. Por otro lado, no se alcanza el grado de realismo de los modelos basados en captura de movimientos, pero el

modelo propuesto presenta la ventaja de no estar limitado por un conjunto finito de movimientos capturados, además de no requerir el hardware y software necesario para realizar la captura de movimientos.

Por último, reflexionando sobre la aplicación práctica del modelo propuesto, es obvio que va dirigido al sector de la animación y los diferentes mercados objetivo relacionados, tales como la industria de los videojuegos o el cine de animación. Juegos basados en la música como *Guitar Hero* o películas de animación de Disney en la que son habituales las escenas de baile son cada vez más populares. Para los diseñadores y animadores de estos productos, la capacidad para generar de forma casi automática una extensa variedad animaciones coreográficas con la que dotar a los personajes supone un ahorro tanto temporal como económico y les permite explorar nuevas posibilidades de comportamiento. Por otro lado, atendiendo a los estudios recientes sobre los beneficios de los sistemas audiovisuales como complemento en tratamientos médicos para la reducción del dolor [189] [34], y conectando con los alentadores resultados del experimento llevado a cabo usando el sistema *Melomics Music Medicine* presentado en el capítulo 2, queda abierta la integración del modelo propuesto en este sistema para la generación automática de contenidos audiovisuales destinados a tratar patologías en el ámbito médico.



## Capítulo 4

### Desarrollo artificial aplicado a búsqueda de forma y comportamiento creativos de agentes escaladores

En este capítulo, se describe un modelo de desarrollo artificial inspirado en el formalismo de las gramáticas de grafos reguladas por cadenas, inicialmente propuesto por Lobo et al. [115], el cual implementa una codificación indirecta para abstraer el proceso de desarrollo regulado por el genoma. El modelo desarrollado ha sido integrado en un algoritmo evolutivo, y se ha aplicado a la búsqueda de formas y comportamiento en agentes u organismos virtuales 3D, que deben encontrar solución a la tarea de escalar por una pendiente dentro de un entorno virtual físicamente realista. A diferencia de los modelos propuestos en los capítulos 2 y 3, que estaban orientados a la producción automática de contenidos audiovisuales, el presente modelo explota las capacidades creativas de los algoritmos evolutivos para el diseño tanto de forma como de control y comportamiento de estructuras de carácter ingenieril, concretamente agentes virtuales, con potencial aplicación en el campo de la robótica. En este sentido, el objetivo consiste en liberar al diseñador o ingeniero de tener que tomar decisiones sobre todos los detalles del diseño de un robot escalador. Aunque parte de los diseños obtenidos en las simulaciones realizadas resulten complicados de implementar físicamente, éstos podrían servir de inspiración a ingenieros y diseñadores en la adopción de soluciones. El uso de codificación indirecta inherente en el formalismo de las gramáticas de grafos reguladas por cadenas, en conjunción con la búsqueda evolutiva, permite la generación de una gran variedad de formas y comportamientos, a partir de genomas relativamente sencillos. Los resultados de las simulaciones muestran que comportamientos relativamente complejos surgen a partir de sistemas de control simples, mediante la explotación de las dinámicas de interacción entre cuerpo y

sistema de control, lo que permite enmarcar el modelo presentado dentro del concepto de computación morfológica. Por último, ante la hipótesis de que organismos con simetría bilateral podrían desempeñar mejor la tarea de escalar por una pendiente, se compararon los resultados de dos versiones del modelo, uno en el que se favorecía de forma más intensa la simetría bilateral de los organismos desarrollados y otro en el que no se favorecía este tipo de diseño.

Este capítulo está organizado de la siguiente manera. En la sección 4.1, se hace una breve introducción a los fundamentos de las gramáticas de grafos reguladas por cadenas y una revisión de los trabajos previos sobre co-evolución de morfología y control de agentes; y sobre las aplicaciones de robots escaladores. En la sección 4.2, se describe el modelo propuesto, mientras que en la 4.3 se muestran los resultados de las simulaciones llevadas a cabo. Por último, en la Sección 4.4, se presentan las conclusiones derivadas del modelo propuesto y sus aplicaciones.

## 4.1. Introducción y trabajo relacionado.

El modelo propuesto utiliza un proceso de desarrollo inspirado en el formalismo de las gramáticas de grafos reguladas por cadenas, que es evolucionado para optimizar la búsqueda de forma y comportamiento de agentes, con aplicación en el diseño de robots escaladores. Por este motivo, en esta sección se describe este innovador formalismo y se presenta una breve revisión de los trabajos previos presentes en la literatura sobre co-evolución y desarrollo de morfología y control de agentes; y las diferentes aplicaciones de los robots escaladores.

### 4.1.1. Las gramáticas de grafos reguladas por cadenas

El formalismo de las gramáticas de grafos reguladas por cadenas fue propuesto por Lobo et al. [115], y ha sido aplicado con éxito en problemas de búsqueda de

#### 4.1. Introducción y trabajo relacionado.

---

comportamiento [113] y animación automática de personajes [112]. Este formalismo abstrae el proceso de desarrollo regulado por el genoma. Por un lado, una cadena de símbolos representa la secuencia del genoma, por otro, la gramática de grafos abstrae las acciones que se producen durante el proceso de desarrollo de un organismo. De esta forma, un organismo se representa por un grafo, de modo que cada arista representa una célula en el organismo y cada nodo representa la conectividad entre las células. Las reglas de la gramática representan las acciones que una célula puede efectuar durante el desarrollo, tales como divisiones. De esta manera, un proceso de desarrollo corresponde con una derivación gramatical, donde el axioma (zigoto) es derivado mediante la aplicación de un conjunto de producciones sobre las aristas (las acciones de las células durante el desarrollo) reguladas por una cadena (el genoma). La regulación de la derivación gramatical mediante una cadena de símbolos está inspirada en la división celular biológica. De la misma manera que el genoma de un organismo se propaga a través de sus células durante su desarrollo, la cadena reguladora se propaga a través de las aristas del grafo en cada paso de reescritura. El método de desarrollo del modelo propuesto en este capítulo toma inspiración de este formalismo con algunas salvedades y modificaciones que serán descritas en las secciones siguientes.

##### **4.1.2. Evolución y desarrollo de morfología y control en agentes**

En los últimos años una de las principales líneas de investigación en computación evolutiva y vida artificial ha sido el estudio de la emergencia de formas y comportamientos diversos mediante modelos basados en agentes. Una estrategia comúnmente utilizada consiste en modelar estos agentes de una forma detallada simulándolos en entornos físicamente realistas. El crecimiento de las capacidades de computación permitió el uso de métodos evolutivos para explorar la emergencia de formas y comportamientos de los agentes. El trabajo de Sims es un ejemplo clásico (que se puede considerar adelantado a su tiempo): la simulación de una población de

agentes, inmersos en un entorno físicamente realista, permitió obtener una gran diversidad de morfologías y comportamientos o estrategias, que evolucionaban mediante el uso de funciones de bondad que premiaban a los mejores agentes en tareas de locomoción [161] y de competencia [160]. Del mismo modo, dentro del campo de la robótica, la disciplina de la robótica evolutiva implica no sólo optimizar el sistema de control o comportamiento de un agente robótico, sino también su forma o estructura corporal. Debido a que esta co-evolución de forma y control incrementa el número de variables a optimizar, el espacio de búsqueda crece desmesuradamente, por lo que se han utilizado estrategias de codificación indirecta [165] (desarrollo de cuerpo y sistema de control a partir de una especificación genética compacta). Este es el caso de los trabajos de Bongard [23], Hornby, Lipson y Pollack [89], [91] y de otros autores que usan algoritmos que simulan el desarrollo [6] para optimizar robots en entornos virtuales simulados. La fabricación aditiva semiautomática ha sido empleada para construir versiones físicas de los robots evolucionados de forma simulada [110]. Algoritmos evolutivos también han sido utilizados para permitir a los robots defectuosos recuperarse de sus averías [25], así como conseguir una fácil transferencia de los controladores evolucionados en los robots simulados a su versión física [104]. Además de la locomoción, algunos autores han evolucionado comportamientos más exigentes, tales como discriminar entre objetos de diferentes formas mediante su manipulación [176] o nadar en un fluido [95]. También se ha evolucionado el comportamiento para robots con diseños corporales no tradicionales como la locomoción de robots con tensegridades [136], *soft-robotics* [145] y robots en enjambre [151].

En casi todos estos modelos, el sistema de control es bastante complejo (a menudo, una especie de red neuronal recurrente). Sin embargo, se ha demostrado que en muchos casos es innecesaria esta complejidad [135], [121], [113]. En este sentido, la computación morfológica [135] es una metodología de diseño para los agentes robóticos, la cual explota la dinámica de la interacción entre el cuerpo y el sistema de control del agente, con el fin de mantener este sistema de control tan simple como

sea posible. Las aplicaciones van desde robots bípedos semi-pasivos con sistemas de control mínimos [121], a los robots de tensegridad cuya compleja dinámica no-lineal es aprovechada para generar un patrón de locomoción con un control mínimo [135]. Los seguidores de camino de Lobo y Vico [113] representan un ejemplo extremo de la computación morfológica, ya que el control es puramente mecánico, sin ningún tipo de modelo abstracto de control. El modelo presentado en este capítulo sigue esta línea, y se enmarca dentro de la computación morfológica utilizando un proceso de desarrollo de agentes robóticos virtuales basado en codificación indirecta, que es inmerso en un algoritmo evolutivo para co-evolucionar formas y comportamiento.

La robótica puede beneficiarse en gran medida de la capacidad creativa mostrada por los algoritmos evolutivos para encontrar soluciones de diseño y comportamiento innovadores. Cuando se aplica así, un enfoque evolutivo puede liberar al ingeniero de tener que tomar decisiones sobre todos los detalles de diseño del robot. En muchos casos, el algoritmo evolutivo puede descubrir soluciones que el diseñador podría no haber pensado, sobre todo para aquellos robots que no tienen un diseño o sistema de control intuitivo para un ser humano. Por ejemplo, a menudo es difícil ver la mejor manera de controlar un robot de cuerpo blando utilizando técnicas de aprendizaje automático tradicionales, y mucho menos determinar la mejor combinación de materiales blandos y rígidos para su diseño.

Por último, como se mencionó en el capítulo 1, en la implementación de un algoritmo evolutivo adquiere una gran importancia el esquema de codificación genotipo-fenotipo. Los algoritmos evolutivos tradicionales utilizan un método de codificación directa (es decir, el individuo se representa por una fila o *array* de datos que está vinculada de forma directa a sus propiedades). Sin embargo, una asignación directa se ha demostrado no ser eficaz en problemas complejos, debido a las limitaciones en escalabilidad, adaptabilidad y capacidad de evolución. Por este motivo, los algoritmos evolutivos que utilizan una codificación indirecta genotipo-fenotipo se han aplicado en una amplia gama de problemas [117], [90], [113]. Los

agentes u organismos virtuales modelados en este capítulo son codificados indirectamente mediante un genotipo que describe su proceso de desarrollo.

### 4.1.3. Aplicaciones de los robots escaladores

En la literatura podemos encontrar una gran variedad de trabajos que recogen la utilización y adecuación de robots escaladores en diferentes tareas y contextos. Por ejemplo, en el caso de la industria nuclear, los robots con capacidad para escalar paredes son muy apreciados, especialmente para trabajar en ambientes de atmósfera contaminada. Este es el caso del robot SCOUR, o del robot SADIE [186] que realiza inspección en conductos. La industria de la construcción también se beneficia de la utilización de este tipo de robots [140] para realizar tareas de mantenimiento, tales como limpiar cristales en las fachadas de edificios, como los robots SFR II [184], RACCOON [155] y Sky Cleaner [193]; o ROBOCLIMER [128], un robot cuadrúpedo portador de herramientas y maquinaria. De forma similar, la industria civil utiliza robots escaladores para pintar las estructuras de puentes metálicos [175]. El robot REST en sus diferentes versiones, utilizado para pintar y soldar el casco de barcos; y el robot MACS I, que inspecciona el fuselaje de aviones, representan aplicaciones de robots escaladores en la industria naval y aeronáutica, respectivamente. El proyecto conocido como vehículo de reconocimiento (*CRV*) o *cliffbot*, es un ejemplo actual del uso de robots escaladores, destinado a la búsqueda de vida en Marte gracias a su capacidad para trepar empinados acantilados y montañas marcianas, llegando a nuevos lugares aún por explorar. Por otra parte, la naturaleza ha servido de fuente de inspiración en el diseño de robots escaladores. Un claro ejemplo de ello es la imitación del sistema de adhesión de los dedos de las extremidades del gecko [7] para su aplicación en vigilancia y sistemas militares. El diseño y comportamiento de los agentes virtuales obtenidos con el modelo propuesto en este capítulo podría aplicarse potencialmente para el diseño de robots escaladores similares a los presentados en esta sección.

### 4.2. Descripción del modelo

El problema que se pretende abordar es el diseño de la forma y comportamiento de agentes u organismos virtuales 3D compuestos por elementos tales como barras (*struts*), resortes elásticos, motores y sensores, que sean capaces de ascender por una pendiente con una inclinación de 45°. Tanto el organismo como el escenario han sido simulados utilizando el motor físico *ODE*. En esta sección se describe un modelo muy simple de la expresión genética que permite el desarrollo de un organismo de este tipo. El modelo está inspirado en el formalismo de las gramáticas de grafos reguladas por cadenas, pero extendido con propiedades geométricas. En una primera instancia nos centraremos en explicar cómo se codifica la estructura de los organismos, mientras que la física de la simulación se describirá más adelante. También se propone una estrategia alternativa al modelo descrito para su posterior comparación en la sección 4.3.

#### 4.2.1. Proceso de desarrollo artificial

El modelo propuesto sigue los principios del desarrollo artificial usando una codificación indirecta del genoma para abstraer formalmente el proceso de desarrollo, que constituye una correspondencia compleja entre genotipo y fenotipo. Más concretamente, un organismo se desarrolla a partir de una única célula o cigoto de acuerdo a la información contenida en su genoma, mediante reescritura guiada por una gramática de grafos regulada. El sistema de producción de la gramática está formado por reglas que pueden alterar las propiedades de los nodos y aristas del grafo, así como hacer que un nodo se reproduzca y dé lugar a dos nodos hijos conectados a él a través de nuevas aristas. El genoma de un individuo se representa mediante una cadena que resulta de la concatenación de subcadenas, conteniendo cada una una regla, y los valores numéricos o parámetros que instancian los

atributos la regla. Estas subcadenas pueden considerarse como genes, ya que representan unidades indivisibles de la expresión genética. La Figura 4.1a muestra una interpretación del modelo en términos biológicos. Representando el organismo por un grafo tridimensional etiquetado, un nodo corresponde a una célula en este esquema, y las aristas que los conectan son medios estructurales de adhesión a las células vecinas. Esto supone una diferencia con respecto al trabajo inicial de Lobo y Vico [114], en el que una célula se correspondía con una arista del grafo y los nodos como puntos de unión entre ellas. Cada célula almacena una copia del genoma y expresará un determinado gen en un momento dado del proceso de desarrollo. Este dominio de expresión formado por el conjunto de genes del genoma que una célula expresará, se implementa como un par de marcadores (*Inicio*, *Fin*) para los genes, que indican el comienzo y el final del dominio. Además de la transformación del grafo que puede producir, la aplicación de una regla siempre altera el dominio de una célula. En cada paso de derivación, cada célula expresa simultáneamente el primer gen de su dominio. Si la expresión de este gen determina que la célula se reproduzca y de lugar a otras dos células descendientes, el resto del dominio de la célula progenitora se transmite a los descendientes por igual y la célula progenitora deja de expresar su dominio, pasando éste a ser la cadena vacía. En caso de que el gen no suponga creación de células descendientes, la célula actual continuará expresando el resto de su dominio en los siguientes pasos de derivación. Si el dominio de una célula llega a ser la cadena vacía, entonces la célula ha terminado de expresar el genoma. El desarrollo de un organismo se detiene cuando todas las células terminan de expresar el genoma, ocurriendo esto siempre en un tiempo finito, como se muestra más adelante.



## 4.2. Descripción del modelo

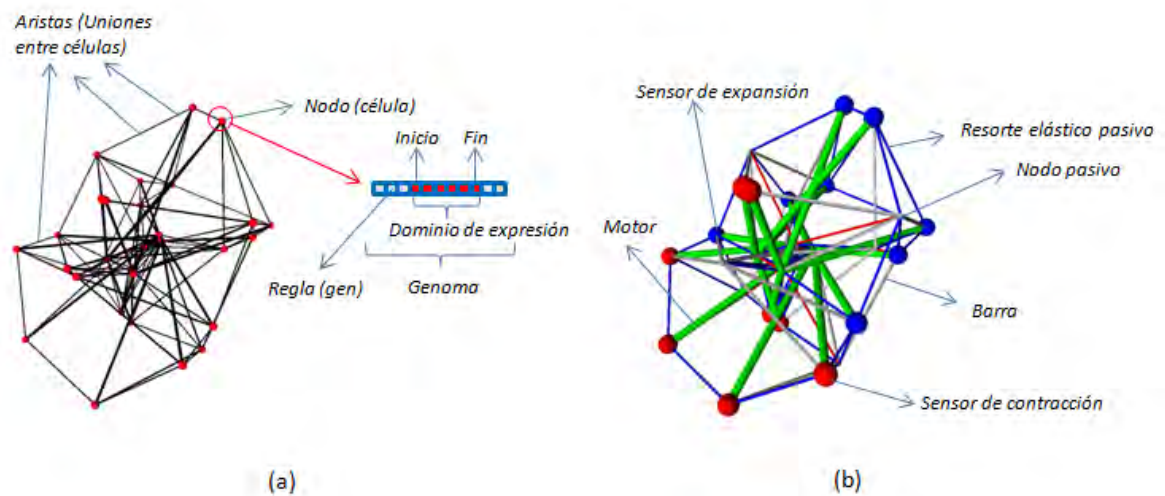


Figura 4.1: Correspondencia entre grafo desarrollado mediante una gramática regulada por cadenas y organismo funcional que representa. (a) Grafo desarrollado a partir de un único nodo (célula) que expresa diferentes reglas (genes) de su dominio de expresión contribuyendo a la creación de nuevos nodos (color rojo) y aristas (color negro). (b) Funcionalidades y propiedades físicas de los elementos del organismo desarrollado representado por el grafo. Los nodos representan nodos pasivos (color gris) y sensores (color azul o rojo); y las aristas representan barras (color gris), resortes elásticos pasivos (cilindros delgados de color rojo, azul o verde) y motores (cilindros gruesos de color verde, azul o rojo).

A fin de obtener una morfología 3D que represente un organismo capaz de interactuar en un entorno virtual simulado, el grafo se extiende en un espacio euclídeo tridimensional donde cada nodo posee coordenadas espaciales  $(x, y, z)$  y funcionalmente podrán representar nodos pasivos y nodos sensores. Por su parte, las aristas que unen los nodos representarán tres tipos de elementos: barras físicas (*struts*), resortes elásticos pasivos y motores. Las reglas de la gramática pueden afectar las propiedades de estos elementos (nodos y aristas) con el fin de configurar la morfología y funcionalidad del organismo desarrollado. La Figura 4.1b muestra las propiedades físicas y funcionales de los diferentes elementos que componen un organismo. Los individuos siempre comienzan el desarrollo a partir de un grafo inicial que es el axioma de la gramática (o cigoto), y que está formado por un único nodo o célula. El proceso de reescritura continúa hasta que el desarrollo se detiene.

El organismo desarrollado está entonces listo para ser simulado en un entorno virtual en el que deben enfrentarse al reto de ascender por una pendiente, como se describe en la siguiente sección. Las tres reglas del modelo generativo se describen a continuación, existiendo una regla para implementar la reproducción de una célula dando lugar a dos descendientes, otra para transformarla en sensor y otra para crear aristas motoras.

#### 4.2.1.1. La regla de reproducción (R).

Esta regla determina que la célula progenitora se reproduzca y de lugar a otras dos células descendientes equidistantes con respecto al progenitor, y que estarán unidas a éste mediante barras o resortes elásticos. Esta regla está instanciada por cuatro parámetros que determinan la dinámica espacial y mecánica de esta reproducción. El primer parámetro  $l$ , toma valores dentro del rango  $[0.1, 9.9]$  e indica la distancia, con respecto al progenitor, a la que se crean las dos células descendientes. Gráficamente, las tres células se sitúan en una línea imaginaria con el progenitor en el centro y los descendientes en cada uno de los extremos a una distancia  $l$  del progenitor (Figura 4.2c). El vector que determina la dirección de esta línea imaginaria viene determinado al girar el vector unitario  $\vec{u}$  el ángulo  $\theta \times \pi$ , definido en radianes, sobre el eje de coordenadas *axis*. El vector unitario  $\vec{u}$  es un valor internamente almacenado en el progenitor cuyo valor es  $(0, 0, 1)$  para la célula inicial (Figura 4.2a). El parámetro  $\theta \in [0.1, 0.9]$ , indica el ángulo  $\theta \times \pi$  que el vector  $\vec{u}$  gira sobre el eje determinado por el parámetro *axis*  $\in \{1, 2, 3\}$ , donde 1 corresponde con el eje  $x$ , 2 con el eje  $y$ , 3 el eje  $z$  (Figura 4.2b). El vector unitario resultante de esta transformación,  $\vec{u}'$ , indica la dirección de la línea imaginaria en la que se sitúan los dos descendientes y es almacenado internamente en cada descendiente para ser utilizado en una futura reproducción de éstos, de la misma forma que  $\vec{u}$ . El cuarto

## 4.2. Descripción del modelo

---

parámetro de la regla,  $tipo \in \{0, 1\}$ , indica el tipo de elemento que representa la arista con la que los descendientes se unen físicamente al progenitor, siendo un resorte elástico en el caso de  $tipo=0$  (Figura 4.2e) o una barra en el caso de  $tipo=1$  (Figura 4.2d). En cuanto al grado de libertad de las uniones barra-nodo, estas uniones son fijas en el punto de unión entre la barra y el nodo progenitor; y son de tipo bisagra (*hingjoint*) en el punto de unión entre la barra y los nodos descendientes, siendo  $\bar{u} \times \bar{u}'$  el eje sobre el que se realiza el giro con una amplitud  $\pi/4$  en cada sentido (Figura 4.2g). En el caso de reproducciones de segundo nivel o superior, es decir, cuando el progenitor ha sido generado a su vez por otra célula en otra etapa anterior del proceso de desarrollo, los nodos descendientes se unirán al predecesor de su progenitor mediante resortes elásticos (Figura 4.2f). Por último, en cada división ambos descendientes heredan de forma idéntica el dominio de expresión del progenitor (excepto el gen de reproducción que acaba de expresar). Con la intención de establecer una ligera diferenciación celular en cada descendiente, las células mantienen un valor interno  $spin \in \{-1,1\}$  heredado en cada reproducción celular y cuyo valor es 1 para la célula inicial. En cada reproducción, un descendiente hereda el valor  $spin$  del progenitor multiplicado por -1 y el otro descendiente el valor de  $spin$  del progenitor multiplicado por 1 (Figura 4.2h). Este simple mecanismo de diferenciación determinará el tipo de sensor (ver regla  $\mathcal{S}$ ) si las células descendientes expresan el gen  $\mathcal{S}$  en posteriores etapas del proceso de desarrollo. En resumen, la regla de reproducción (R) consta de los siguientes parámetros  $\{l, \theta, axis, tipo\}$ . Un ejemplo de esta regla puede ser R2.00.511, con  $l=2.0$ ,  $\theta=0.5$ ,  $axis=1$  y  $tipo=1$ .

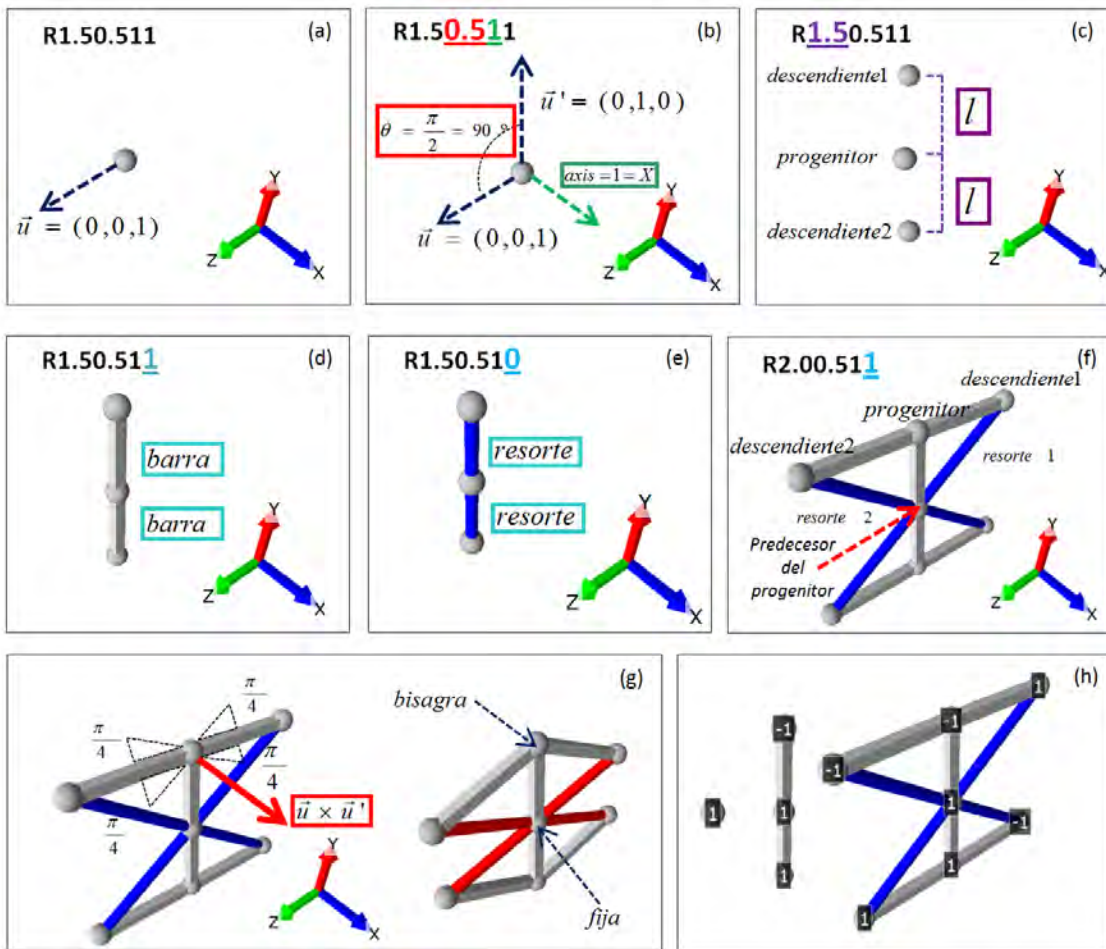


Figura 4.2: Regla de producción  $R$ . (a) Vector interno  $\vec{u}$  del progenitor. (b) Vector  $\vec{u}'$  que indica la dirección espacial de la línea sobre la cual se crean los descendientes. Viene determinado al girar  $\theta \cdot \pi$  radianes el vector  $\vec{u}$  del progenitor alrededor del eje  $axis$  (c) Distancia  $l$  a la que se crean los descendientes (d) Unión progenitor-descendientes mediante barras  $tipo=1$  (e) Unión progenitor-descendientes mediante resortes  $tipo=0$  (f) Unión predecesor de progenitor-descendiente mediante resortes (g) Grados de libertad de las uniones barra-nodo (h) Distribución de los valores  $spin$  durante reproducciones.

#### 4.2.1.2. Regla motor ( $M$ ).

Cuando esta regla forma parte del dominio de expresión de una célula, y es interpretado antes de una regla  $R$ , especifica que en esta próxima reproducción de la célula los resortes elásticos pasivos que se crean se convertirán en resortes elásticos activos (motores) que podrán ser de contracción o de expansión dependiendo de si

## 4.2. Descripción del modelo

los nodos unidos a ellos son sensores y del tipo de éstos (controlado por la regla  $S$ ). En el caso de que, en la reproducción, el progenitor se una a los descendientes mediante dos resortes, éstos se convertirán en motores y las uniones predecesor del progenitor-descendiente (en caso de existir) serán resortes elásticos pasivos. Si por el contrario, la unión progenitor-descendiente se realiza mediante barras, las uniones predecesor del progenitor-descendiente (si existen) serán los motores. El único parámetro de esta regla,  $f \in [0.1, 1.0]$ , determina la fuerza con la que actuará el motor cuando el sensor lo requiera. En la práctica, el parámetro  $f$  modifica temporalmente la longitud de reposo del resorte provocando la contracción o expansión del mismo. La relación entre los genes  $R$  y  $M$  es un ejemplo sencillo de regulación genética del genoma, ya que por un lado la expresión de  $M$  está condicionada a que exista un gen  $R$  en una posición posterior del dominio de expresión de la célula, y por otro, la presencia de  $M$  afecta al modo en el que  $R$  se expresa. La Figura 4.3 muestra la manera en la que la presencia (a) o no (b) de un gen  $M$  en el dominio de expresión, afecta al modo de expresión de un gen  $R$ . Un ejemplo de este tipo de regla puede ser  $M0.5$ , con  $f=0.5$ .

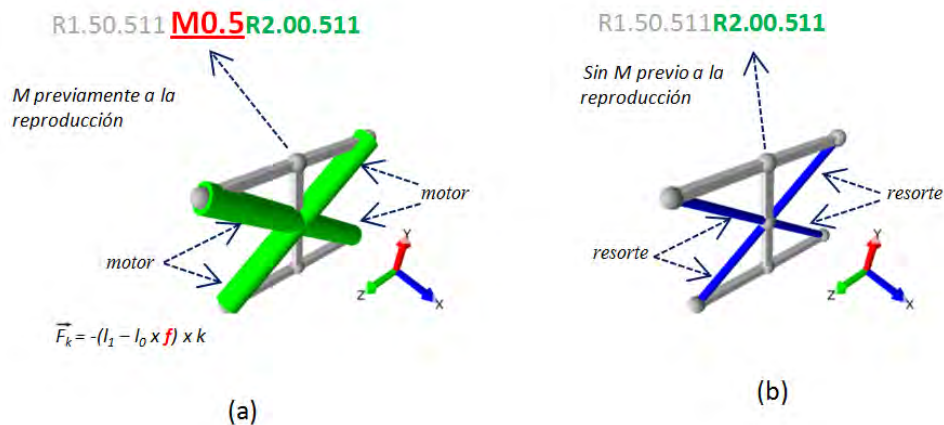


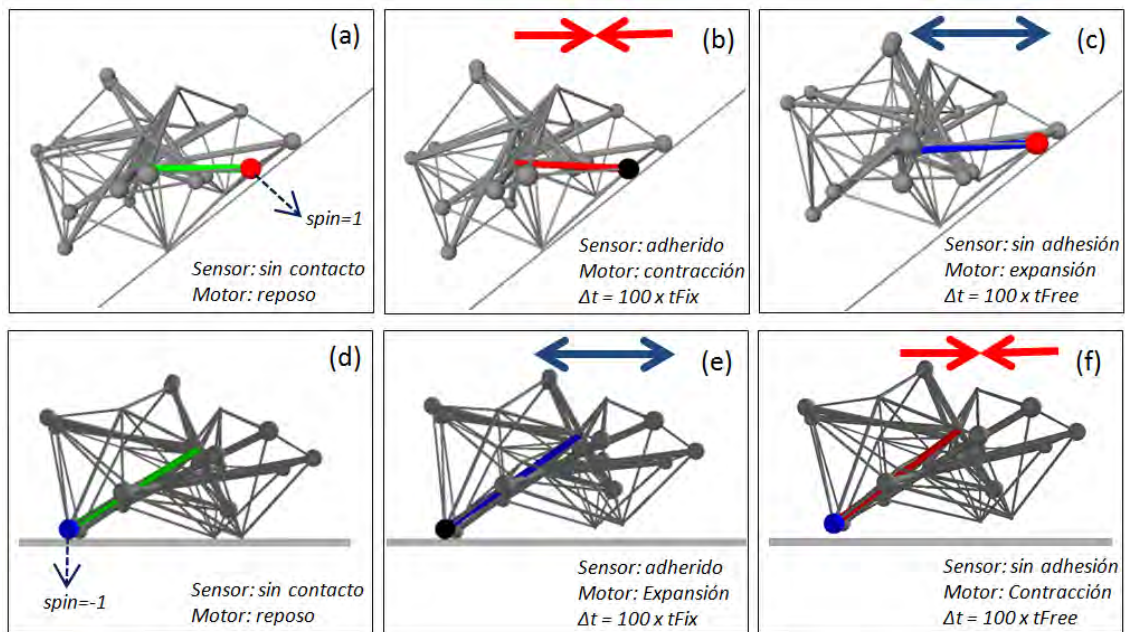
Figura 4.3: Regla de producción  $M$ . (a) Presencia de  $M$  antes de  $R$  en el dominio de expresión de la célula: cuando la célula exprese  $R$  se crearán motores en vez de resortes elásticos pasivos. (b) Reproducción ( $R$ ) sin presencia previa de  $M$ : la reproducción no dará lugar a la creación de motores.

#### 4.2.1.3. La regla sensor (*S*).

Esta regla determina si un nodo actuará como sensor y define sus propiedades sensoriales, así como el tamaño de éste en proporción a su tamaño actual. Con el objetivo de cumplir el reto al que se debe enfrentar el organismo desarrollado, ascender por una pendiente, un nodo sensor también actúa a modo de ventosa adhiriéndose a la superficie por la que debe escalar. Existen dos tipos de sensores: de contracción o de expansión. Un sensor de contracción (color rojo), al detectar contacto con la superficie, se adhiere a ella durante un tiempo determinado por el parámetro  $tFix$  e indica a los motores unidos a él que se contraigan durante este tiempo  $tFix$ . Transcurrido  $tFix$  el sensor se despega de la superficie ya que pierde su capacidad de adherencia durante un tiempo indicado por el parámetro  $tFree$ , y los motores se expanden durante ese tiempo. Pasado  $tFree$ , el sensor vuelve a recuperar su capacidad de adherencia y los motores conectados a él pasan a estado de reposo, pudiendo de nuevo operar como se ha descrito. La Figura 4.4a-c muestra el funcionamiento de un sensor de contracción. Por el contrario, un sensor de expansión (color azul), al contactar con la superficie, se adhiere a ella durante un tiempo  $tFix$  e indica a los motores unidos a él que se expandan durante  $tFix$ , tras el cual el sensor se despega de la superficie ya que pierde su capacidad de adherencia durante  $tFree$  y los motores se contraen. Al igual que los sensores de contracción, pasado el tiempo  $tFree$ , el sensor vuelve a recuperar su capacidad de adherencia y los motores conectados a él pasan a estado de reposo. La Figura 4.4d-f muestra el funcionamiento de un sensor de expansión. El rango de valores tanto para el parámetro  $tFix$  como para  $tFree$  es  $[0.1, 1.0]$ . Traducido este intervalo de valores a pasos de simulación, el tiempo de adherencia y libertad puede variar entre 10 y 100 pasos de simulación. Visto de un modo funcional, los sensores de contracción se pegan a la superficie en un punto y pretenden acercar el resto del organismo a ese punto durante  $tFix$  intentando encontrar un punto más avanzado al que fijarse durante  $tFree$ , mientras que los sensores de expansión operan igual pero alejando el organismo de un punto. El resultado es la emergencia de un comportamiento

## 4.2. Descripción del modelo

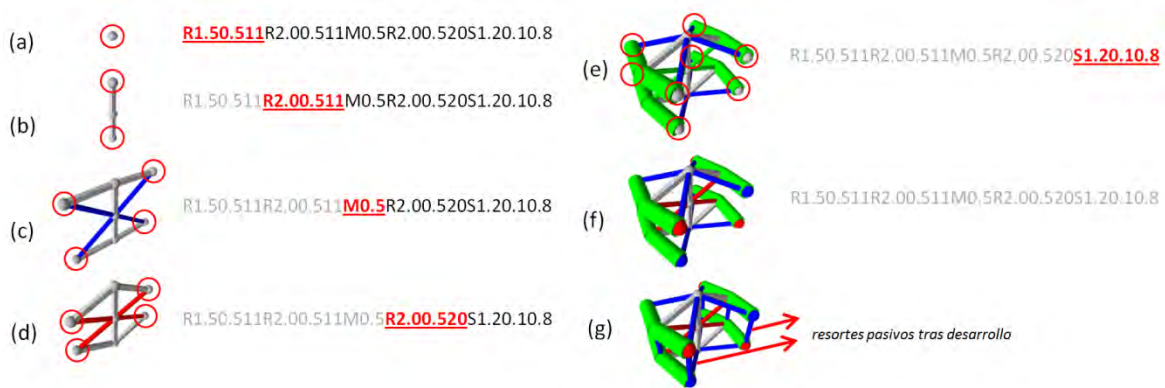
locomotor en el organismo en una dirección y sentido determinados con capacidad para ascender por una rampa gracias a las cualidades de adherencia de los sensores. El tipo de sensor viene determinado por el valor interno de la célula  $spin \in \{-1,1\}$ , de forma que es de contracción (color rojo) si  $spin=1$  y de expansión (color azul) si  $spin=-1$ . Por último, la expresión del gen  $S$  también puede afectar al tamaño de éste en proporción a su tamaño actual. Esta modificación viene determinada por el parámetro  $s \in [0.5, 4.0]$ . En resumen, la regla sensor ( $S$ ) consta de los siguientes parámetros  $\{tFix, tFree, s\}$ . Un ejemplo de esta regla puede ser  $S0.50.13.8$ , con  $tFix=0.5$ ,  $tFree=0.1$  y  $s=3.8$ .



**Figura 4.4: Regla de producción S. (a-c) Funcionamiento de un sensor de contracción. (d-f) Funcionamiento de un sensor de expansión.**

Un vez que todas las células han expresado el genoma y el proceso de desarrollo ha terminado y, con el objetivo de conferir más estabilidad y rigidez al organismo, se lleva a cabo un proceso de reconexión de nodos. Para ello se calcula la triangulación tridimensional de *Delaunay* de todos los nodos, y se unen mediante resortes elásticos pasivos aquellas parejas de nodos que cumplan (1) no existe previamente ningún

tipo de arista o elemento (barra, resorte elástico, motor) que los conecte y (2) la distancia entre el par de nodos sea inferior a la longitud media de elementos elásticos (resortes elásticos o motores) conectados a ambos nodos. Si un par de nodos cumple esas condiciones se crea un resorte elástico pasivo entre ellos cuyas propiedades físicas (longitud de reposo, constantes de resorte y amortiguación) es calculado según la media de los elementos conectados a ambos nodos. El resultado es un organismo virtual con una estructura más estable y volumétrica, que de acuerdo a las pruebas experimentales realizadas, los hacen más ideales para ser simulados en un entorno virtual sometido a leyes físicas.



**Figura 4.5: Proceso de desarrollo de un organismo simple desarrollado a partir del genoma  $R1.50.511R2.00.511M0.5R2.00.520S1.2$  desde (a) el zigoto, hasta el organismo maduro (g)**

La Figura 4.5 ilustra como un genoma de ejemplo  $R1.50.511R2.00.511M0.5R2.00.520S1.2$  da lugar a un organismo virtual 3D sencillo. La célula inicial contiene un dominio de expresión que cubre todo el genoma y comenzará a expresar el primer gen de este dominio (Figura 4.5a). La expresión del primer gen R da lugar a la creación de dos descendientes cuya distancia, dirección espacial y modo de unión al progenitor viene determinado por los parámetros  $l=1.5$ ,  $\theta=0.5$  ( $90^\circ$ ),  $axis=1$  (eje  $x$ ) y  $tipo=1$  (barra). Al expresar un gen R el dominio de expresión del progenitor se vacía y es heredado por ambos descendientes por igual, los cuales expresarán el segundo gen R (Figura 4.5b). Esta



## 4.2. Descripción del modelo

---

expresión provoca una nueva reproducción dando lugar a cuatro nuevos descendientes (dos por cada progenitor). Nuevamente el dominio de expresión de las dos células progenitoras se vacía y es heredado por sus cuatro descendientes quienes interpretarán el gen  $M$  (Figura 4.5c). Como se ha comentado, el gen  $M$  no se expresa por sí mismo por lo que no produce ningún tipo de cambio a nivel morfológico en el organismo. El cambio de estado de los resortes pasivos (de expansión a contracción) y la orientación de las barras que se aprecia en la Figura 4.5d es debido a las dinámicas físicas de los elementos que se recalculan en cada paso de iteración durante el desarrollo. Tras interpretar un gen  $M$ , las cuatro células siguen interpretando su dominio de expresión, pasando a expresar el gen  $R$  (Figura 4.5d), que provoca la creación de nuevos descendientes (ocho en total) que estarán unidos a sus progenitores mediante motores y no simples resortes pasivos debido a la interpretación previa por los progenitores del gen  $M$ . Estos nuevos descendientes heredan el dominio de expresión de sus progenitores y pasan a interpretar el gen  $S$  (Figura 4.5e). Este gen transforma las ocho células en sensores modificando su tamaño y asignándole sus propiedades sensoriales. De acuerdo al valor interno *spin* los sensores serán de contracción (rojos) o expansión (azules) (Figura 4.5f). En este punto el desarrollo regulado por el genoma se ha completado, ya que todas las células tienen un dominio de expresión vacío. Finalmente, se produce un proceso de reconexión de nodos para inferir más estabilidad y rigidez al organismo (Figura 4.5g), y poder ser sometido a simulación dentro de un entorno virtual físicamente realista.

### 4.2.2. Variación del modelo: dominio de expresión heredado variable

Como se ha mencionado, en la reproducción celular, el dominio de expresión heredado por ambos descendientes es el mismo, permitiendo que regiones idénticas del genoma puedan ser expresadas simultáneamente en diferentes partes del organismo, favoreciendo y fomentando la simetría bilateral. Con el objetivo de

constatar que esta simetría bilateral favorece el buen desempeño de los organismos virtuales en el reto de escalar por una pendiente, se ha llevado a cabo una pequeña variación del modelo en el que, en la regla de reproducción celular, el dominio de expresión restante del progenitor no se hereda completamente por ambos descendientes, sino que se divide en diferentes proporciones mediante un quinto parámetro  $p \in [0.2, 0.8]$  que determina qué porción del dominio del progenitor es heredado por cada descendiente. Por ejemplo, ante una célula progenitora con un dominio de expresión formado por  $n=5$  genes,  $p=0.4$  indica que el primer descendiente hereda los  $g1 = p \times n = 0.4 \times 5 = 2$  primeros genes del dominio; y el segundo descendiente hereda los  $g2 = n - g1 = 5 - 2 = 3$  últimos genes. De esta forma, se reduce considerablemente la posibilidad de obtener organismo con simetría bilateral, aunque en ocasiones pueda obtenerse debido a las características inherentes del modelo. La Figura 4.6 muestra las diferencias entre la regla de reproducción del modelo original y la variación alternativa propuesta. La comparación de resultados de ambas estrategias será mostrada en la sección 4.3. La diversidad de formas y comportamientos que se pueden obtener usando este sistema de generación descrito (en sus dos variantes) es lo suficientemente grande como para cubrir las necesidades de la mayoría de las estrategias búsqueda de formas y comportamientos.



Figura 4.6: Comparación de comportamiento del gen  $R$ . (a) la estrategia del modelo original de fomento de simetría bilateral y (b) en la estrategia alternativa de no fomento de simetría.

### 4.2.3. Física del modelo

Como se ha comentado, los organismos desarrollados son inmersos en un entorno virtual regido bajo leyes físicas, y deben resolver el problema de ascender una pendiente. En este sentido, con el fin de alcanzar un grado aceptable de realidad, se ha utilizado el motor físico *ODE*, tanto para la construcción de los organismos como para el entorno virtual. Como se ha señalado en el apartado anterior, las aristas representan barras y resortes elásticos, que pueden ser pasivos o activos (motores). La dinámica de los resortes se determina por la constante del resorte, la constante de amortiguamiento, y la longitud en reposo, siendo los dos primeros globales para todos los resortes, y el último particular para cada uno de ellos. Estos valores configuran las propiedades de compresión y tracción del resorte. El hecho de que un resorte se estire o se comprima en un momento dado durante la simulación depende de los parámetros y las fuerzas aplicadas sobre el mismo de forma interna (en el caso de los motores), y de su interacción con el entorno y resto de elementos del organismo. Todos los resortes en el organismo comparten las mismas constantes de resorte y amortiguación ( $k$  y  $c$ , respectivamente), mientras que la longitud de reposo se establece durante el desarrollo. La dinámica del resorte resulta de aplicar estas ecuaciones sobre los nodos que une:  $\vec{F}_k = -k\vec{l}$  (ley de Hook), y  $\vec{F}_c = -c\vec{l}$  (fuerza de amortiguación), donde  $\vec{l}$  es el vector de desplazamiento del resorte (es decir, la distancia y la dirección en la que se deforma el resorte). Durante el desarrollo, la dinámica de los resortes es evaluada, y contribuye a la forma final de la estructura (el fenotipo).

Por otro lado, los organismos están compuestos también por barras y nodos (implementados como esferas) ambos con características físicas tales como masa y densidad. Cuando el organismo es colocado en el entorno virtual para enfrentarse al reto de subir una pendiente, estos elementos se ven afectados por la fricción con el medio. Esto se implementa de acuerdo con la expresión  $\vec{F}_m = -\mu_m \vec{v}$ , donde  $\mu_m$  es el

coeficiente de fricción, y  $\vec{v}$  es el vector de velocidad del nodo o barra. Aparte de estas fuerzas, la simulación también incluye la fuerza gravitatoria  $\vec{F}_g = -g\vec{u}_z$ , donde  $g$  es la aceleración constante de la gravedad, y  $\vec{u}_z$  es el vector ortonormal al eje Z. Con el fin de reducir el coste computacional de las simulaciones, se ha desactivado la detección de colisiones entre elementos de los organismos, no así entre los elementos de los organismos y el resto del entorno como el suelo o pendiente. La eliminación de detección de colisiones entre los elementos del organismo conlleva a que, en la práctica, algunos de los diseños simulados sean complicados de diseñar físicamente.

#### 4.2.4. Algoritmo genético

Las dos versiones del modelo propuesto (tanto la versión inicial que fomenta la simetría bilateral como la alternativa que no favorece este diseño), han sido integradas en un algoritmo genético para la búsqueda forma y comportamiento en organismos que sean capaces de subir por una pendiente con inclinación de 45°. Cada individuo de la población se desarrolla a partir de su información genética, y después es inmerso en un mundo virtual físicamente realista. El valor de bondad de un individuo representa cómo de bien consiguió ascender por una pendiente, determinado como el cociente de dividir la distancia en línea recta desde el punto inicial y el punto final (*trayectoriaMínima*) entre la trayectoria real seguida (*trayectoriaReal*), multiplicado todo ello por la altura alcanzada al final de la simulación. La siguiente ecuación combina estas magnitudes en un único valor:

$$f(s_k) = \left( \frac{\text{trayectoriaMínima}}{\text{trayectoria Real}} \right) \bullet h$$

donde  $f(s_k)$  es el valor de bondad para el organismo  $k$  de la población que se ha simulado. Cada organismo es simulado durante  $t=10000$  pasos de tiempo. El

## 4.2. Descripción del modelo

---

numerador se calcula midiendo la distancia entre el punto inicial  $p1$  y el punto final  $p2$ .  $p1$  se corresponde con el centro de masa del organismo cuando se encuentra en el suelo antes de comenzar la simulación, y  $p2$  es el centro de masa del organismo al final de la simulación. El denominador se calcula mediante la suma de las distancias entre centros de masa cada  $t=100$ . El valor  $b$  representa la altura a la que se encuentra el organismo al final de la simulación y coincide con el valor de la coordenada  $y$  del centro de masa del organismo. Como resultado, la función resultante calcula un valor de bondad más alto para aquellos organismos que asciendan por la pendiente la mayor altura posible y lo hagan en línea recta, desviándose lo menos posible. Esto coincide, por lo general, con una ascensión más rápida en menos tiempo.

Se sometió a evolución a dos poblaciones (una para cada estrategia propuesta del modelo) de tamaño fijo de 300 individuos durante 200 generaciones. Los individuos de la población inicial fueron genomas con un tamaño entre 4 y 10 genes, siendo los genes y los valores de parámetros generados aleatoriamente. Los individuos que forman las nuevas poblaciones se obtienen mediante la selección por ruleta de forma directamente proporcional a su valor de bondad, y posterior mutación de los seleccionados, de acuerdo con tres operadores: (1) Inserción: un nuevo gen al azar se inserta en una posición aleatoria; (2) Borrado: un gen se elige aleatoriamente y se elimina; y (3) Modificación de atributo: un atributo de un gen se selecciona aleatoriamente y se sustituye por otro valor aleatorio del atributo. Las probabilidades asociadas a cada operador de mutación se han ajustado a los valores 0.22, 0.15, y 0.35, respectivamente, lo que fomenta la modificación e inserción de nuevos genes y, en menor medida, su eliminación. El elitismo de individuos no ha sido implementado.

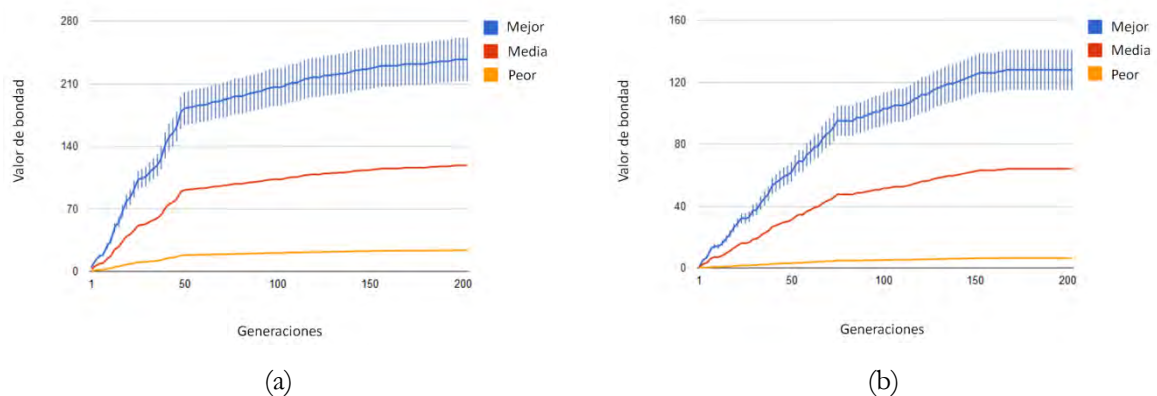
### 4.3. Resultados obtenidos en las simulaciones

Tanto para el desarrollo del modelo como para la implementación del algoritmo genético se ha utilizado el lenguaje de programación *Python*, debido a que permite crear prototipos con rapidez y dispone de una gran cantidad de librerías de simulación física y animación. Con el objetivo de obtener datos suficientes para comprobar la validez representativa de ambas versiones del modelo, se ejecutó el algoritmo genético un total de 50 veces (25 veces para cada versión). Para reducir el tiempo total de ejecución del algoritmo, se paralelizaron las simulaciones de los 300 individuos de cada generación en un clúster de 320 CPUs (2 GHz). De media, cada ejecución del algoritmo (incluyendo inicialización y evolución) conllevó 12 horas aproximadamente.

En la Figura. 4.7 se puede observar las curvas de valor de bondad promedio en las 25 ejecuciones utilizando las versiones (a) fomento de simetría bilateral (b) no fomento de simetría bilateral. Las curvas de color azul indican la evolución de la media del valor de bondad del mejor individuo de cada generación, representando las barras de error la desviación estándar de este valor en las 25 ejecuciones de cada versión del modelo. Las curvas rojas y amarillas representan el valor de bondad medio promedio, y el peor valor de bondad en promedio, respectivamente. En una primera observación, se deduce que cuando se fomenta la simetría, se obtienen valores considerablemente más elevados para el valor de bondad. Esto confirma la hipótesis inicial sobre que el fomento de la simetría favorece que los organismos desarrollados cumplan con el reto de subir por una pendiente con mejores resultados. Cuando se fomenta la simetría se aprecia una convergencia más rápida de las soluciones obteniéndose un incremento rápido del valor de bondad hacia la generación 50 siendo el valor medio de bondad del mejor individuo igual a 192. A partir de este momento el modelo converge de una forma más lenta. Por el contrario, cuando no se fomenta la simetría, la convergencia se produce con mayor

### 4.3. Resultados obtenidos en las simulaciones

lentitud, siendo 92 el valor medio de bondad del mejor individuo en la generación 82. A partir de este momento se produce un ascenso casi lineal y alrededor de la generación 162 se produce un estancamiento en la evolución. En lo referente a la creación de estructuras con simetría bilateral, utilizando la estrategia de fomento de simetría, se obtuvieron un 87% de estructuras que presentaban simetría bilateral, por tan solo un 16% usando la estrategia de no fomento de simetría. Vistos los resultados, se puede concluir por lo tanto que cuando se fomenta de forma intensa la simetría de los organismos, estos cumplen mejor su tarea aunque los resultados obtenidos cuando no se fomenta esta simetría disten de ser pobres.



**Figura 4.7: Promedio del valor de bondad del mejor individuo, medio y peor para las 25 ejecuciones llevadas a cabo para cada versión del modelo (a) fomento intenso de la simetría (b) no fomento de la simetría.**

La Figura 4.8 muestra el desarrollo de tres de los mejores organismos evolucionados usando la estrategia de fomento de la simetría bilateral. El color de los resortes elásticos pasivos (cilindros más finos) indica su estado de tensión interna, siendo los resortes de color rojo resortes en compresión y los azules en estado de expansión. El color verde se usa para denotar a los resortes en estado de reposo o equilibrio (ni en compresión ni en expansión). Los motores son representados por cilindros más gruesos utilizando el mismo código de colores que los resortes pasivos. Los nodos sensores son coloreados en azul (sensor de expansión) y rojo (sensor de contracción). Las morfologías de los organismos desarrollados con esta estrategia se caracterizan por sus formas regulares y simétricas. Como se comentó anteriormente,

esto es debido al hecho de que en la reproducción celular, el dominio de expresión que heredan ambos descendientes es el mismo permitiendo que regiones idénticas del genoma puedan ser expresados simultáneamente en diferentes partes del organismo, favoreciendo y promoviendo la simetría.

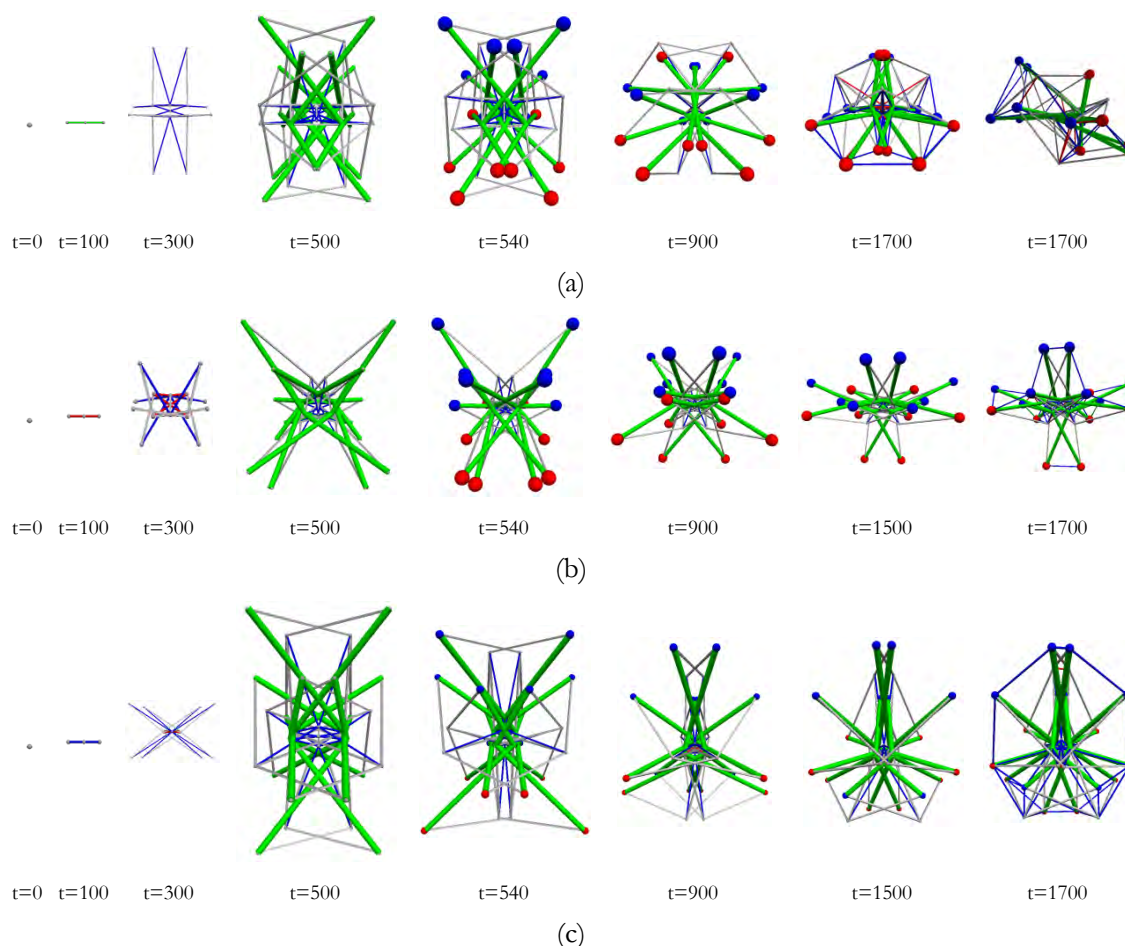


Figura 4.8: Distintas etapas del proceso de desarrollo de tres de los mejores organismos obtenidos con la estrategia de fomento de simetría corporal. Para cada organismo, la imagen más a la izquierda representa el cigoto y la más a la derecha el organismo completamente desarrollado. El color de los resortes elásticos pasivos (cilindros delgados) muestra su estado de tensión. Los motores (cilindros gruesos) aparecen en color verde indicando que están en reposo (no activos). Los nodos sensores son coloreados en azul (sensor de expansión) y rojo (sensor de contracción).

De izquierda a derecha, cada fila se corresponde con un hito durante el proceso de desarrollo del organismo, que empieza con la célula inicial cigoto y termina con el



### 4.3. Resultados obtenidos en las simulaciones

fenotipo u organismo completamente desarrollado. La regularidad y la simetría se puede observar desde las etapas iniciales del desarrollo.

De igual modo, la Figura 4.9 muestra el desarrollo de tres de los mejores organismos evolucionados usando la estrategia de no fomento de la simetría bilateral. Las morfologías de los organismos desarrollados con esta estrategia se caracterizan por sus formas asimétricas, debido a que en la reproducción, los descendientes no heredan el mismo dominio de expresión por parte del progenitor, por lo que ambos expresarán y heredarán a sus descendientes diferentes dominios de expresión con diferentes genes en cada uno de ellos. Aunque con esta estrategia puede surgir la

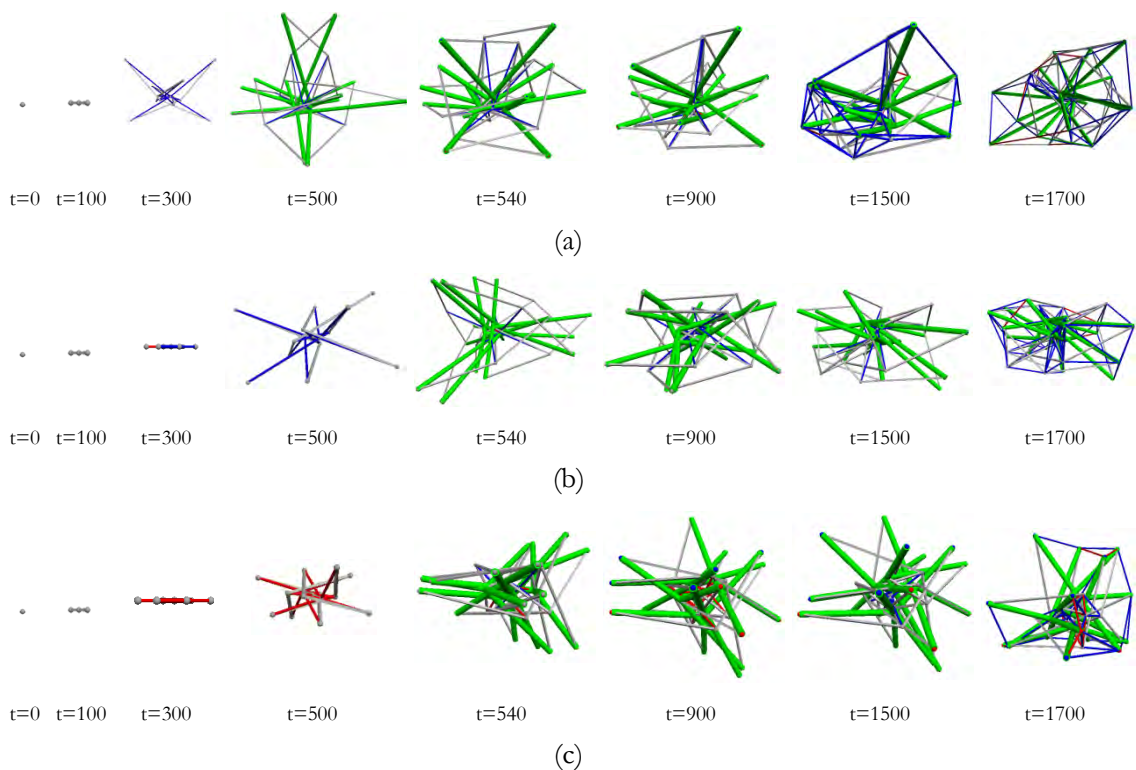


Figura 4.9: Distintas etapas del proceso de desarrollo de tres de los mejores organismos obtenidos con la estrategia de no fomentar la simetría corporal. Para cada organismo, la imagen más a la izquierda representa el cigoto y la más a la derecha el organismo completamente desarrollado. El color de los resortes elásticos pasivos (cilindros delgados) muestra su estado de tensión. Los motores (cilindros gruesos) aparecen en color verde indicando que están en reposo (no activos). Los nodos sensores son coloreados en azul (sensor de expansión) y rojo (sensor de contracción).

simetría, la probabilidad de que esto ocurra se reduce considerablemente con respecto a la estrategia anterior en la que se potencia esta característica. Como puede apreciarse, los organismos no presentan regularidad y ni simetría desde etapas muy tempranas de su desarrollo.

Instantáneas de la simulación del mejor organismo evolucionado con cada estrategia se muestran en las Figuras 4.10 y 4.11, respectivamente. Tras terminar su proceso de desarrollo, los organismos son colocados en un entorno virtual en la posición espacial inicial (0, 0, 0) (Figuras 4.10a, 4.11a). El reto al que se deben enfrentar es ascender por una pendiente inclinada representada como un plano de 45° de inclinación. Debido a que a priori la dirección y sentido que tomará el organismo es desconocida, éste es “encerrado” entre cuatro pendientes formando una especie de embudo como puede apreciarse más claramente en la Figura 4.11i. El número de pasos de simulación de los que disponen los organismos para completar esta tarea es de  $t=10000$ . Con respecto al mejor individuo obtenido siguiendo una estrategia de fomento de la simetría (Figura 4.10) puede apreciarse como consta de dos sensores de contracción delanteros simétricamente dispuestos (al igual que toda su morfología) y que realizan las labores de brazos propulsores. También dispone de sensores de contracción y expansión laterales (visibles en la Figura 4.10e) que le ayudan a recobrar la verticalidad en caso de vuelco; y de sensores de expansión traseros que impulsan a la estructura (Figura 4.10b) y complementan la labor de los sensores de contracción delanteros en el proceso de locomoción y comienzo de ascensión de la pendiente. Morfológicamente, este individuo ha evolucionado para reducir el número de nodos en contacto con la superficie durante el proceso de ascensión, utilizando sólo los dos sensores de contracción delanteros y dos nodos pasivos estructurales (Figura 4.10c) que hacen la labor de apoyo de la estructura del individuo. Este mínimo contacto con la superficie reduce la fricción y simplifica en gran medida el mecanismo de control del individuo durante el proceso de ascensión.

### 4.3. Resultados obtenidos en las simulaciones

---

Básicamente el sistema de control de este individuo simula ligeramente la forma en la que un ser humano intentaría subir por una superficie ayudándose sólo de sus dos brazos. En la Figura 4.10d, se puede observar que mientras se sujeta con la mano del brazo izquierdo a la superficie (sensor en color negro), intenta estirar el brazo derecho (motor en color azul) para alcanzar un punto más alto al que sujetarse. Cuando lo alcanza, suelta el brazo izquierdo y lo estira para alcanzar un punto paralelo al que está sujetado el brazo derecho (Figura 4.10e) y así sucesivamente. Trasladado a la estructura del individuo las manos serían representadas por los sensores y los brazos por el conjunto motor-sensor. El resultado de la co-evolución de esta morfología simétrica y sistema de control simplificado, permite al individuo realizar una ascensión en línea recta que se desvía tan sólo  $22^\circ$  de la perpendicular con la recta que marca la cima (Figura 4.10i), alcanzando un valor de bondad de 269. Por otro lado, en la Figura 4.11 se puede apreciar la simulación del mejor individuo desarrollado usando la estrategia no fomento de la simetría. Como se puede observar tiene una morfología totalmente asimétrica en la que apenas se aprecia regularidad (Figura 4.11a). La disposición irregular de sensores y motores contribuye a que el organismo no pueda realizar la labor de locomoción y ascensión siguiendo una trayectoria óptima (Figura 4.11e), además de hacerlo con mayor lentitud que el individuo anterior. No obstante, lejos de lo que pueda aparentar, este individuo consigue ascender por la pendiente manteniendo una línea recta al principio aunque desviada unos  $42^\circ$  de la perpendicular. En parte esto es debido a que aprovecha la esquina de unión entre dos pendientes. Sin embargo, llegando al final de la simulación comienza a desviarse describiendo una pequeña curva, posiblemente al abandonar la esquina y no ofrecer un buen rendimiento al pasar a una sola superficie.

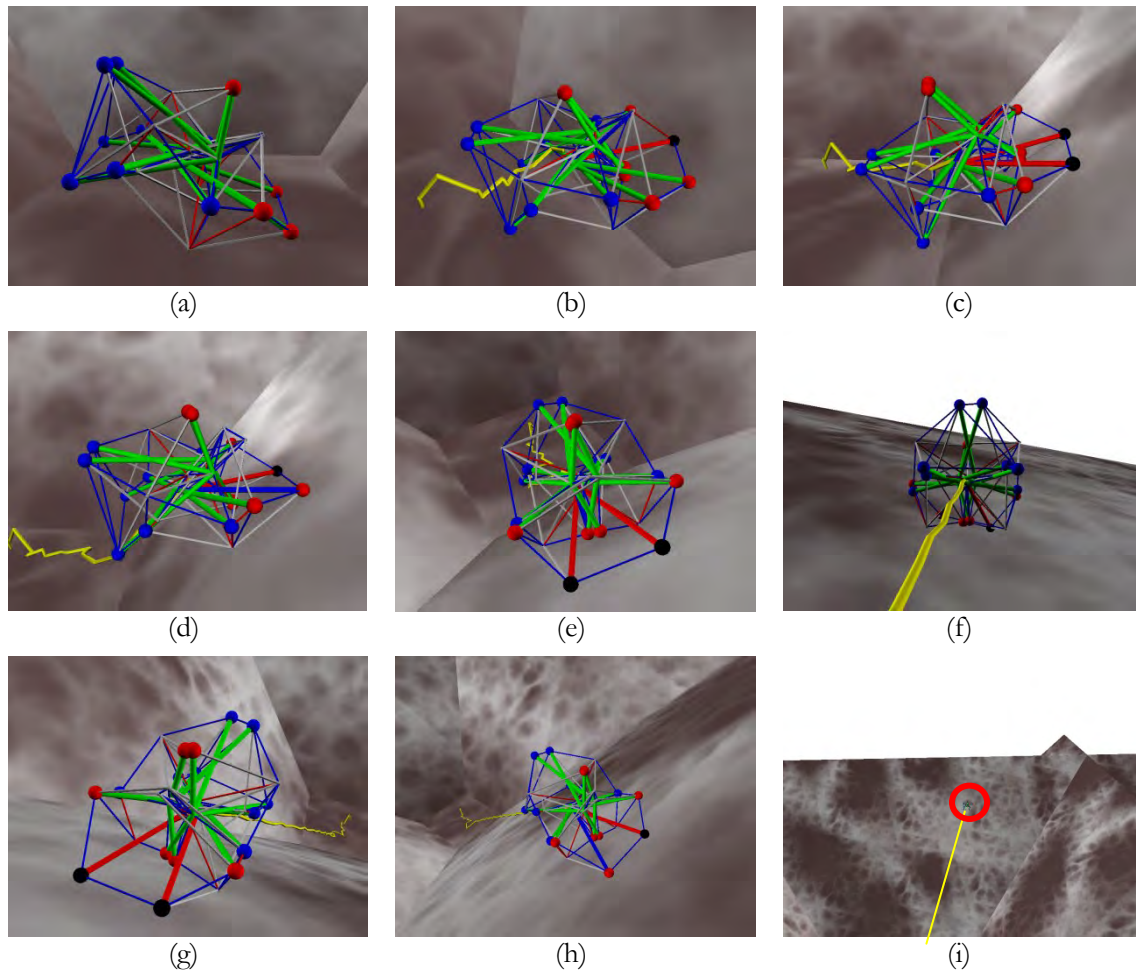


Figura 4.10: Simulación del individuo con mejor valor de bondad (269) desarrollado usando la estrategia de fomento de la simetría. La línea amarilla indica la trayectoria seguida por el individuo durante la tarea de escalar por una pendiente. (a) Tras completar el desarrollo, el individuo es colocado en la posición espacial inicial (0, 0, 0). (b) El individuo se aproxima a la pendiente y comienza la ascensión. (c) Individuo en plena ascensión sujetándose a la superficie con dos sensores delanteros (color negro). (d) Instante en el que se aprecia cómo se sujeta con un sensor (color negro) y éste contrae el motor unido a él. Mientras el otro “brazo” delantero se expande intentando alcanzar un punto más alejado al que sujetarse. (e) Vista frontal del individuo. (f) Vista trasera del individuo. (g) Zoom del individuo en su proceso de ascensión (h) Instantánea del individuo en un momento avanzado de la ascensión. (i) El individuo se encuentra cerca de alcanzar la cima. Se puede apreciar como la trayectoria seguida (línea amarilla) dibuja una línea recta que se desvía relativamente poco (22°) de la perpendicular con la recta que marca la cima.

### 4.3. Resultados obtenidos en las simulaciones

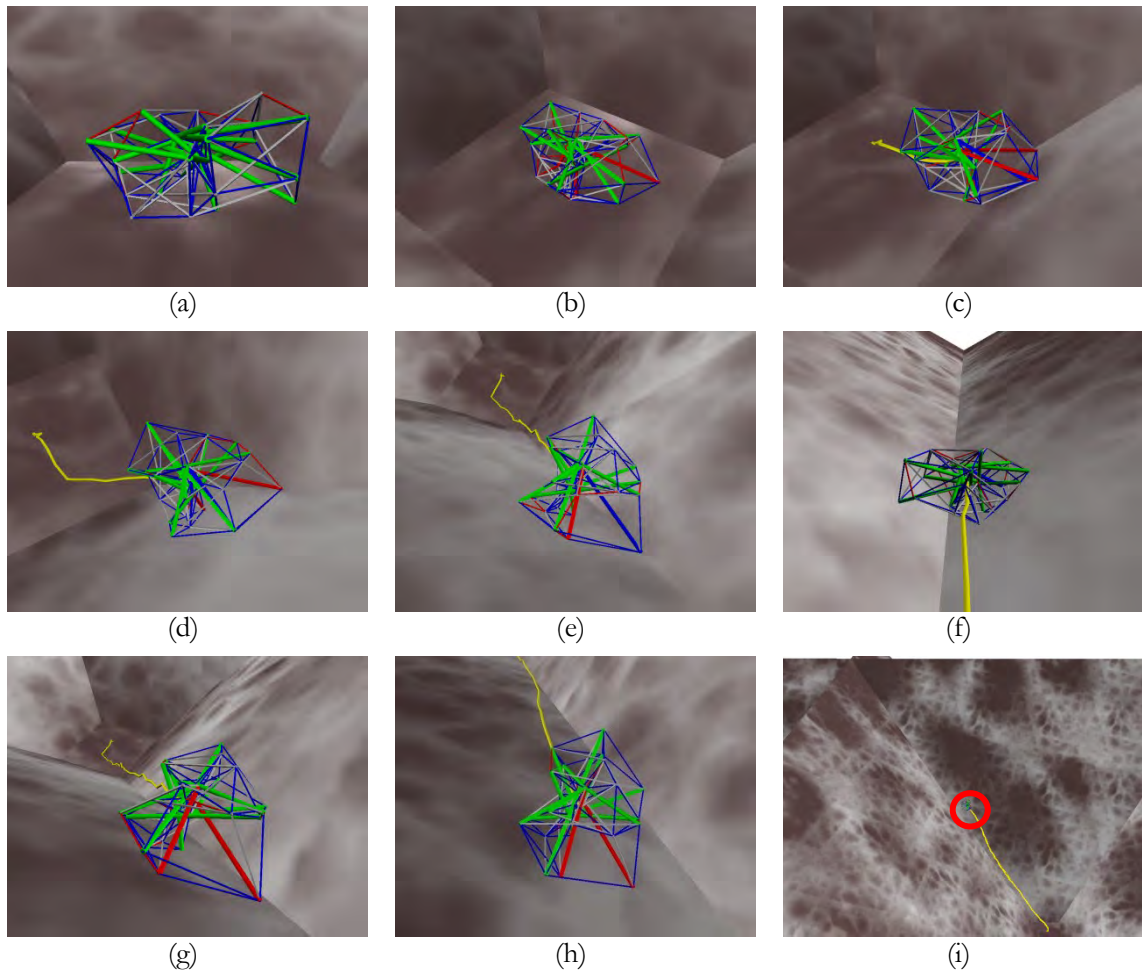


Figura 4.11: Simulación del individuo con mejor valor de bondad (138) desarrollado usando la estrategia de no fomento de la simetría. La línea amarilla indica la trayectoria seguida por el individuo durante la tarea de escalar por una pendiente. Los sensores existen pero no han desarrollado su tamaño lo suficiente para ser apreciados visualmente. (a) Tras completar el desarrollo, el individuo es colocado en la posición espacial inicial (0, 0, 0). (b) El individuo se aproxima a la pendiente caminando mediante la interacción de los sensores y motores con el suelo. (c) El individuo comienza la ascensión sujetándose a la superficie con dos sensores delanteros y contrayendo sus correspondientes motores (d) Instante en el que se aprecia cómo se sujeta con un sensor (color negro) y éste contrae el motor unido a él. (e) Vista frontal del individuo. (f) Vista trasera del individuo. (g) Zoom del individuo en su proceso de ascensión (h) Imagen del individuo en un momento avanzado de la ascensión. (i) Instantánea del individuo cerca de agotar el tiempo de simulación ( $t=9500$ ). Se puede apreciar como la trayectoria seguida (línea amarilla) dibuja una línea recta al principio con una ligera curvatura al final desviándose en aproximadamente  $42^\circ$  de la perpendicular con la recta que marca la cima.

## 4.4. Conclusiones y discusión

En este capítulo se ha presentado un modelo de desarrollo artificial inspirado en el formalismo de las gramáticas de grafos reguladas por cadenas destinado a la búsqueda de formas y comportamientos creativos. Los resultados de las simulaciones contribuyen a confirmar la idoneidad del uso de este formalismo, que implementa una codificación indirecta, como sistema de desarrollo de estructuras siguiendo la línea de los trabajos previos [114], [113]. Concretamente, el modelo desarrollado se ha aplicado a la co-evolución de formas y comportamiento de organismos virtuales que deben cumplir la tarea específica de escalar por una pendiente. El fomento del surgimiento de propiedades biológicas tales como la simetría bilateral y regularidad se han demostrado muy útiles al ser adoptadas por los organismos. De acuerdo con los principios de la computación morfológica, la interacción entre las dinámicas de cuerpo y sistema de control contribuye de manera fundamental a simplificar su sistema de control (Figura 4.10) ofreciendo un desempeño bastante aceptable en el cumplimiento de la tarea. Las ventajas del desarrollo de simetría bilateral también se han puesto de manifiesto al ser comparada con otra estrategia en lo que no se fomentaba de forma especial esta característica de diseño, confirmando el hecho establecido de que la simetría bilateral está directamente relacionada con el óptimo desempeño en tareas de locomoción.

El resultado de las simulaciones de los organismos desarrollados dentro de un entorno virtual físicamente realista, y las propias características físicas y funcionales de los elementos que conforman los organismos (resortes elásticos, barras, motores y sensores) permiten orientar la aplicación de este modelo al desarrollo de robots alineados dentro del campo de la *soft-robotics*, si bien no todos los diseños obtenidos puedan ser implementados físicamente. En este sentido, el modelo propuesto supone una herramienta creativa para la optimización de diseños hardware (forma) y

#### 4.4. Conclusiones y discusión

---

software (sistema de control) de este tipo de robots sirviendo de apoyo al diseñador y liberándolo de tener que tomar decisiones sobre todos los detalles del diseño. En muchos casos, el algoritmo evolutivo descubre soluciones que el diseñador podría no haber pensado, sobre todo para aquellos robots que no se pueden diseñar o controlar de manera intuitiva por un ser humano. El reto evolutivo al que se han enfrentado los agentes, ascender por una pendiente, resulta un escenario relativamente común en el que tendría que desenvolverse un robot de exteriores. Un ejemplo de ello es la inspiración del sistema de adhesión de los dedos de las extremidades del gecko [7] para su aplicación en vigilancia y sistema militares. En misiones de exploración espacial, un robot del tipo que se han propuesto en este capítulo podría aterrizar accidentalmente en un cráter y ser capaz “escapar” de él gracias a la estrategia desarrollada para escalar por superficies inclinadas, al igual que los vehículos de reconocimiento *cliffbot*, aunque sean lanzados desde gran altura ya que los resortes elásticos que lo componen le permiten adquirir una gran capacidad para resistir y amortiguar golpes. Por otro lado, como se ha mencionado en el capítulo 3, la animación por computador es otra disciplina que también se puede beneficiar de las soluciones aportadas por la creatividad computacional. En este sentido, la diversidad de formas y comportamientos que se pueden obtener usando este modelo (en sus dos variantes) resulta bastante amplia y sobre todo, supone un repertorio casi infinito de formas y comportamientos no triviales no sometidos al sesgo o ideas preconcebidas de los animadores, pudiendo ser utilizado en la creación de personajes con comportamientos y movimientos complejos a la vez que realistas, siguiendo la línea de trabajos anteriores [112].





## Capítulo 5

### El desarrollo biológico como inspiración para la creación de agentes multicelulares

La ingeniería embriomórfica, una instancia particular de la ingeniería morfogénica, toma su inspiración directamente del desarrollo biológico, para contribuir al diseño de nuevos sistemas de auto-organización capaces de sustituir a los arquitectos e ingenieros, y sus diseños, por grandes colectividades descentralizadas de agentes. En este capítulo se presenta el potencial de aplicación de la ingeniería embriomórfica a la robótica colectiva y reconfigurable a través de un modelo abstracto 3D, para el desarrollo de organismos multicelulares artificiales denominado *MapDevo3D* (Arquitectura modular mediante desarrollo programable, Figura 5.1). Se trata de superestructuras virtuales en enjambre, o agentes [95], que se desarrollan y presentan un comportamiento en un mundo virtual físicamente realista. Sus cuerpos están compuestos por un gran número de células o agentes, que pueden ser considerados como pequeños módulos robóticos o agentes autopulsados. En cualquier caso, el genotipo específico que las células o agentes comparten, hace que la compleja arquitectura que representa el fenotipo sea modular, programable y reproducible. Por un lado, su textura de aspecto casi continuo y flexible acerca este modelo a los ideales de la *soft-robotics*, tomando inspiración del desarrollo biológico como proceso creativo para la construcción de robots de este tipo. Por otro lado las características y propiedades físico-mecánicas de los organismos desarrollados y simulados en un mundo virtual, y la posibilidad de automatizar su producción permiten extender su aplicación al campo de la animación basada en física, concretamente dentro de la disciplina de la animación de cuerpos blandos o *soft-bodies*.

El presente capítulo está estructurado de la siguiente manera. La sección 5.1 incluye una revisión de los trabajos previos relacionados con el modelo presentado en este

capítulo. La Sección 5.2 introduce y explica el modelo *MapDevo3D*, una extensión del modelo original bidimensional [51] de desarrollo embrionario basado en auto-organización, la formación de patrones, y la regulación genética. A continuación, la sección 5.3 sugiere cómo la función puede surgir de la estructura: examina cómo las morfologías 3D anteriores se vuelven funcionales por la animación endógena e inmersión en un entorno virtual físicamente realista, donde pueden interactuar con objetos, exhibir diversos tipos de comportamiento, y ejecutar tareas. Esto allana el camino hacia una exploración de la evolución sistemática de un espacio genómico del desarrollo, es decir, una agenda *artificial evo-devo*, que se discute en la Sección 5.4. Las conclusiones referentes al trabajo presentado en este capítulo, junto con una breve discusión de sus posibles aplicaciones se reflejan en la sección 5.5.

## 5.1. Introducción y trabajo previo relacionado

Como se ha comentado, en este capítulo se presenta un modelo de ingeniería embriomórfica, disciplina enmarcada dentro de la ingeniería morfogenética, y que toma su inspiración directamente de la *evo-devo*. Las aplicaciones de este modelo de desarrollo evolutivo abarcan desde la creación de agentes robóticos dentro del campo de la robótica evolutiva y *soft-bodied*, hasta el desarrollo de agentes animados cuyas características son propias de la animación de cuerpos blandos. Puesto que en el capítulo de introducción de esta tesis se dedicó un apartado a exponer los principios de la ingeniería morfogenética y embriomórfica, en esta sección se llevará a cabo una introducción y revisión de los trabajos previos relacionados con el modelo presentado desde el punto de vista de la ingeniería morfogenética y la robótica y animación *soft-bodied*.

### 5.1.1. Ingeniería morfogénica

De acuerdo a la literatura, dentro de la ingeniería morfogénica, se identifican cuatro clases de métodos de meta-diseño [56] (Figura 5.2): morfologías que se pueden lograr (1) por “construcción”, donde unos pocos agentes construyen una estructura precisa, relativamente pequeña; (2) por “coalescencia”, donde grandes enjambres de agentes crean ciertos patrones o adoptan formas globales; (3) por “desarrollo”, donde los agentes se añaden de forma recursiva por agregación (pseudo-división) a un grupo de células o semilla inicial; y (4) por “generación”, donde subconjuntos del sistema se transforman o son reemplazados por otros, basados en reglas gramaticales, como es el caso del modelo presentado en el capítulo 4. Naturalmente, estas clases se solapan en cierta medida.

#### 5.1.1.1. Construcción

La robótica modular y la robótica colectiva, dos instancias de la misma idea fundamental según la cuál los sistemas robóticos pueden estar hechos de un número de componentes distribuidos, normalmente se enmarcan dentro de la primera categoría. La robótica modular, o auto-reconfigurable, está interesada en cómo piezas autónomas pero interdependientes pueden reorganizarse para cambiar la estructura general y la morfología de un robot. Por ejemplo, el sistema *M-TRAN* [131], es capaz de desplazarse como un cuadrúpedo o bien reptar, evitar obstáculos, y auto-transformarse dinámicamente a partir de una forma a otra. En otros sistemas modulares, tales como *Molecubes* [194] (Figura 5.2: 1a), piezas con forma de medio cubo giran unas con otras, demostrando auto-reproducción y auto-reparación de morfologías simples, así como evolución (simulada) hacia nuevas formas. La robótica colectiva o de enjambre, por su parte, se centra principalmente en robots móviles individuales que pueden juntarse y formar un sistema más grande, uniéndose entre sí, por medio de abrazaderas o imanes. Por lo general, los *s-bots* [133] (Figura 5.2: 1b) tienen la capacidad de ensamblarse en morfologías apropiadas

y operar como una sola entidad cuando se conectan físicamente. Una lógica de bajo nivel controla las conexiones inter-robot en ciertos ángulos, mientras que una lógica de alto nivel gestiona la secuencia de conexiones hacia morfologías deseadas y la respuesta colectiva adecuada a una tarea.

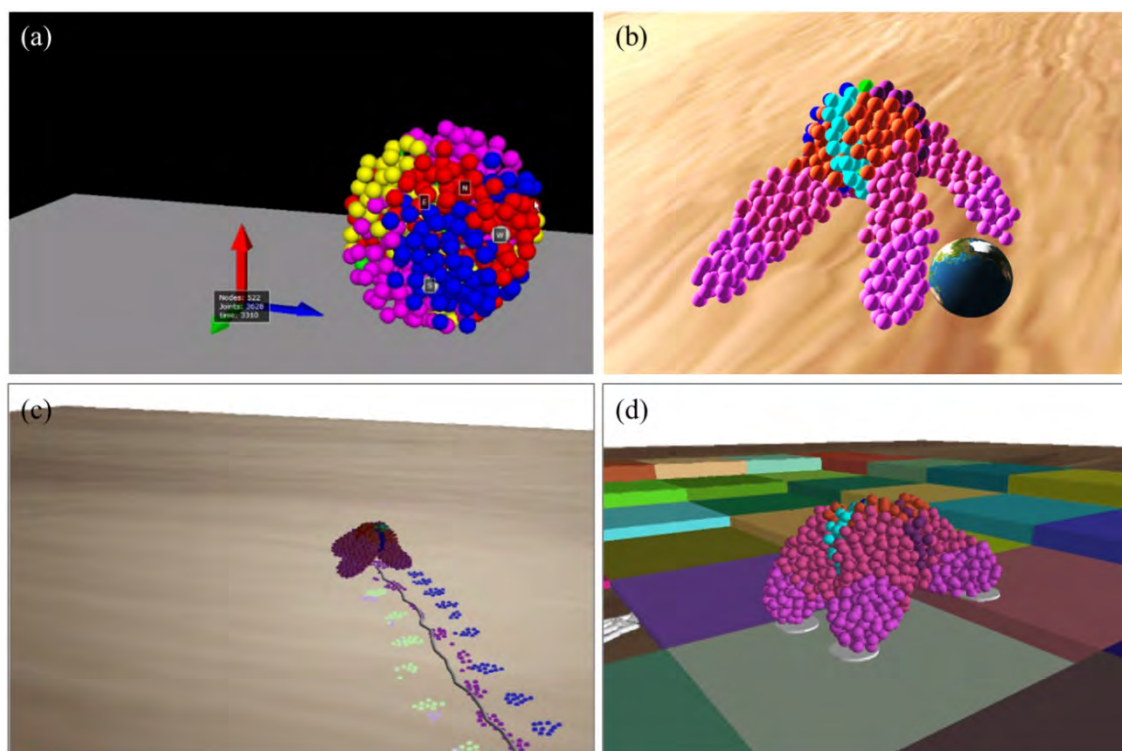


Figura 5.1: Vista previa de los organismos *MapDevo3D* y escenarios. El modelo y los experimentos subyacentes se resumen en este capítulo. (a) Desarrollo del cuerpo de un organismo multicelular, antes del crecimiento de los apéndices, mostrando las regiones de células diferenciadas (Sección 5.2). (b) Organismo maduro con cuatro extremidades largas y delgadas (una de ellas oculta por la perspectiva) caminando sobre el suelo y golpeando una pelota (Sección 5.3). (c) Huellas y traza descrita por el centro de masa de la trayectoria de otro organismo con las extremidades más gruesas. (d) Escenario evolutivo que implica el desafío de trepar por una escalera (Sección 5.4).

Mientras que el límite entre la robótica modular y la colectiva se está difuminando, estos dos dominios todavía se enfrentan al reto de colectivos robóticos auto-organizados, ingenierilmente confiables y funcionales. A menudo, se invierte un considerable esfuerzo en el diseño de hardware sofisticado, especialmente

## 5.1. Introducción y trabajo previo relacionado

actuadores capaces de conseguir un acoplamiento preciso, en detrimento del tamaño del sistema y los principios morfogénéticos de nivel superior. Hay una tendencia en la fabricación de un pequeño número (docenas) de piezas de gran peso, en oposición a la producción masiva de un gran número (cientos) de piezas simples y baratas, e incluso desechables. Como consecuencia de ello, hasta ahora sólo se han conseguido estructuras dispersas hechas de unidades dispuestas en formaciones exactas, tales como cadenas y uniones en T.


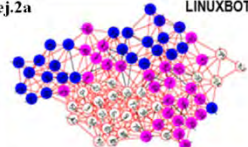
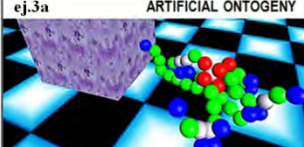
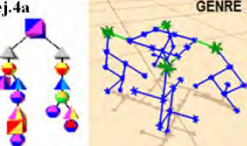


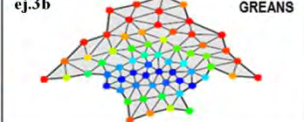
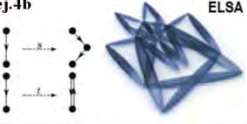
Ingeniería Morfogénica			
Categoría 1: por Construcción	Categoría 2: por Coalescencia	Categoría 3: por Desarrollo	Categoría 4: por Generación
ej.1a  MOLECUBE	ej.2a  LINUXBOTS	ej.3a  ARTIFICIAL ONTOGENY	ej.4a  GENRE
ej.1b  SWARMORPH	ej.2b  KILOBOT	ej.3b  GREANS	ej.4b  ELSA

Figura 5.2: Las cuatro categorías de la ingeniería morfogénica [53], [56], cada una de ellas ilustrada con dos ejemplos de sistemas robóticos, físicamente realizados o simulados. (1) "Construcción": unos agentes construyen una estructura precisa, relativamente pequeña, ya sea robótica modular (ejemplo 1a: Molecubes [194]) o robótica colectiva (ejemplo 1b: Swarmorph [133]); (2) "Coalescencia": enjambres de agentes crean ciertos patrones o adoptan formas globales (ejemplo 2a: Linuxbots [188]; ejemplo 2b: Kilobot [151]); (3) "Desarrollo": agentes se añaden de forma recursiva por agregación (pseudodivisión) a una célula inicial o grupo de células (ejemplo 3a: Artificial Ontogeny [24]; ejemplo 3b: GReaNs [95]); y (4) "Generación": subconjuntos del sistema se transforman o son reemplazados por otros basados en una gramática (ejemplo 4a: GENRE [91]; ejemplo 4b: ELSA [113]).

### 5.1.1.2. Coalescencia

Los sistemas de la segunda categoría de la ingeniería morfogénica contienen un número mucho mayor de agentes móviles, que forman una masa densa o red. Sin

llegar a asociarse, se apiñan permaneciendo cerca unos de otros, y tratan de mantener la comunicación de igual a igual. Aquí, la dinámica de movimiento se inspira generalmente por conceptos químicos tales como feromonas (como en las colonias de hormigas) o morfógenos (como en los tejidos celulares); y sus gradientes de concentración, que son la base para la auto-orientación. Se han propuesto algoritmos de control descentralizados, implementados por ejemplo en *e-pucks* o *Linuxbots*, para vincular la conectividad inalámbrica local con el movimiento del robot a bajo nivel, y crear la “coherencia” global, basados en la agrupación y la conectividad ininterrumpida [188] (Figura 5.2: 2a). En otro avance notable, el proyecto *Kilobot* [151] (Figura 5.2: 2b), cientos de unidades de robots hechos de piezas baratas y rápidas de montar, fueron diseñados específicamente para proporcionar una plataforma de enjambre de gran tamaño en el que poner a prueba los algoritmos de control descentralizados. Sin embargo, a pesar del potencial mayor número de participantes, las habilidades morfogénicas de estos montajes robóticos son aún muy limitados. A diferencia de los sistemas de “construcción” anteriores, ni las conexiones locales, ni la estructura macroscópica son precisas o lo suficientemente fiables. Si bien pueden lograr algún objetivo colectivo funcional, como moverse sin romperse o mostrar patrones simples, los sistemas de coalescencia son demasiado fluidos o demasiado estáticos. En el caso de *Kilobot*, las unidades de hardware, del tamaño de una moneda y con tres agujas vibrantes a modo de hélices son, de hecho, demasiado simplificadas para propósitos morfogénicos.

### 3.2.1.3. Desarrollo

El modelo que se presenta en este capítulo pertenece a la tercera categoría, la cual ofrece lo mejor de ambos mundos: combina las habilidades de auto-ensamblaje precisas de los sistemas de "construcción" con la redundancia y la robustez de los sistemas de "coalescencia". El objetivo es hacer un gran enjambre de agentes que se

unen para formar anatomías macroscópicas reproducibles. Si bien queremos preservar la propiedad esencial de la programación (el enfoque de la ingeniería morfogénica y embriomórfica), también es importante introducir variabilidad y redundancia en el sistema aunque a una escala más pequeña. En el desarrollo biológico, la posición y el número de células individuales es impreciso, mientras que los tejidos y órganos que forman aparecen de forma más precisa. Del mismo modo, el desarrollo artificial puede permitirse ser irregular a nivel microscópico de los agentes individuales, manteniendo una disposición ordenada en los más altos niveles de los grupos de agentes.

Como es natural, la inspiración para esta categoría está cerca de los modelos basados en células de la morfogénesis biológica, y la mayoría de sus modelos podrían ser calificados como (virtuales) *soft-robotics* [144]. Los sistemas parten de un único agente o unos pocos agentes, y crecen hasta un gran tamaño por medio de un proceso repetitivo de división o agregación celular. Los mecanismos de crecimiento incluyen funciones biológicas tales como la señalización molecular y gradientes quimiotácticos. Lo que es más importante, también son controlados por un genotipo (posiblemente una red de regulación genética), que dota a las unidades con la cantidad necesaria de información para tomar decisiones dependientes del contexto entre un rico repertorio de comportamientos posibles. En particular, los sistemas de *artificial ontogeny* [24] (Figura 5.2: 3a) o *artificial embryogeny* [165], [16], [125] han dado paso a un nuevo paradigma en la computación evolutiva (aunque antiguo en la evolución natural) que se basa en asignaciones indirectas de genotipo a fenotipo a través de unas etapas más o menos complejas de desarrollo. En lugar de codificar directamente las características macro del fenotipo (el sistema), el código genético determina las micro características de las células (los agentes), es decir, su capacidad para comunicarse, moverse, y tener afinidad para el ensamblaje con otras células. Imitando a la división celular, la diferenciación y el auto-posicionamiento, un agente genera otros agentes, sigue su propia ruta de ejecución en el programa

genético común (dependiendo de su posición), y crea vínculos específicos con otros agentes de acuerdo con su destino.

Más allá de la estructura corporal y la forma general, sin embargo, lo que en definitiva importa es cómo la criatura desarrollada va a funcionar en un entorno físico (real o simulado). Al igual que en la *artificial ontogeny*, GReaNs [95] (Figura 5.2: 3b) es un modelo de co-evolución paralela de desarrollo y control de movimiento, en organismos *soft-bodies* multicelulares en 2D. El desarrollo es guiado por una red de regulación genética artificial (GRN), y los embriones se convierten en estructuras mediante la conexión de las células vecinas con muelles elásticos. Luego, las células exteriores, que forman la envoltura externa, están sometidas a fuerzas de arrastre en un entorno similar a un fluido. Tanto el programa de desarrollo y el controlador de locomoción, están codificadas por una sola secuencia genómica, que consta de regiones reguladoras y los genes expresados en factores de transcripción y morfógenos. Un algoritmo genético se aplica para evolucionar individuos capaces de nadar en el fluido simulado, donde su valor de bondad depende de la distancia recorrida durante la fase de evaluación. Un trabajo similar en 2D se ha realizado con organismos de mallas de muelles similares a gusanos [156]. El presente capítulo muestra otra combinación de desarrollo y comportamiento de organismos 3D.

#### 5.2.1.4. Generación

En esta última categoría de la ingeniería morfogenética, muy similar a la de “desarrollo”, pero esquemáticamente más inspirada en las plantas que los animales, un sistema es generado por transformaciones sucesivas de sus componentes en el espacio 2D o 3D. Este proceso es controlado por reglas gramaticales, diseñadas a mano o evolucionadas, que tienen el efecto de los componentes de reescritura (es decir, la inserción o eliminación). La familia más popular de estos modelos generativos geométricos, los *L-systems* [139], se originan a partir de una descripción



formal del crecimiento de las plantas a nivel celular. Estos sistemas usan una representación jerárquica basada en cadenas de símbolos agrupadas en corchetes. Este versátil formalismo se utiliza con frecuencia en modelos y simulaciones de crecimiento botánico en ecología teórica [62] o la infografía. En consecuencia, los *L-systems* producen literalmente estructuras arborizadas. Para diversificar las capacidades morfogenéticas de los *L-systems* más allá de topologías fractales, las reglas de reescritura también pueden hacerse dependientes del contexto, es decir, reintroducir algunas interacciones dinámicas y simultáneas entre los componentes que son características del desarrollo, pero que en general están ausentes en los sistemas generativos.

La combinación de los sistemas gramaticales morfogenéticos con la evolución fue probablemente mejor ejemplificada en *Framsticks* [102] y *GENRE* [91] (Figura 5.2: 4a), dos frameworks para el diseño automatizado de robots que caminan, o estructuras estáticas. Desde entonces, la utilidad de los sistemas generativos evolutivos también se ha demostrado en problemas de búsqueda de comportamiento [113] (Figura 5.2: 4b) y grandes estructuras tensegritales [146].

### 5.1.2. Robótica evo-devo y soft-bodied

La robótica *evo-devo* [94], también conocida como *evo-devo-robo*, es una disciplina emergente que básicamente intenta integrar los enfoques evolutivos (robótica evolutiva) y de desarrollo (robótica del desarrollo) para permitir el diseño automático de estructuras robóticas robustas. Éstas se construyen a partir de múltiples componentes más pequeños a través de un proceso de auto-organización que imita el desarrollo de los organismos multicelulares. En tales estructuras, el comportamiento general del cuerpo (como la locomoción) surge como resultado emergente de la cooperación local entre los subcomponentes. En los trabajos iniciales de *evo-devo-robo* [160], las instrucciones genéticas que determinaban el desarrollo fueron codificadas como una clase específica de las gramáticas formales

conocidas como sistemas de Lindenmayer o *L-systems* [139], que se diseñaron inicialmente para modelar el crecimiento de las plantas. Hornby [91] demostró que robots evolucionados usando tales gramáticas de hecho producen formas repetidas. Trabajos adicionales dentro de este campo han sido propuestos en diseño de robots [109], [101], [64], [102], sistemas hardware [163], [77] y *computing tissues* [171]. Dado que el uso de un enfoque *evo-devo* para el diseño de agentes robóticos tiene la ventaja de no estar restringido por los paradigmas de ingeniería existentes, nos permite diseñar automáticamente y de forma completa nuevas clases de agentes robóticos, como por ejemplo robots *soft-bodied* [144]. Los robots de este tipo carecen de uniones o *joints* entre sus partes: en su lugar, todo su cuerpo participa en la locomoción, por ejemplo, mediante la expansión o contracción de sus partes internas, o controlando la rigidez de diferentes áreas de sus cuerpos. La forma del cuerpo al mismo tiempo puede ser arbitraria, y cambiar dinámicamente. Recientemente se han propuesto materiales y métodos que permiten realizar este tipo de locomoción [145], [144], [33]. El campo emergente de la robótica *soft-bodied* ofrece un potencial interesante para crear robots que tendrán capacidades sin precedentes para adaptarse dinámicamente a entornos complejos (por ejemplo, ser capaz de inspeccionar espacios obstruidos cambiando su forma), y al mismo tiempo ser altamente resistentes a los daños y las fuerzas externas debido a su falta de rigidez [174]. Esto dará lugar a aplicaciones en áreas tales como las operaciones de búsqueda o rescate, herramientas de diagnóstico quirúrgico y médico [35] y herramientas de apoyo en la arquitectura [150]. Aunque la capacidad de crear estructuras amorfas con remodelación dinámica abre toda una gama de nuevos modos de locomoción, los paradigmas formales y el conocimiento sobre la mejor manera de diseñar y controlar estructuras tan complejas sólo están comenzando a ser propuestos. Esto ofrece una excelente oportunidad para un enfoque evolutivo y de desarrollo, capaz de crear formas arbitrarias, que nos proporcione nuevas ideas y proponer diseños eficientes, junto con métodos para su control, como han demostrado recientes trabajos preliminares sobre este tema [52].

### 5.1.3. Animación basada en *soft-bodies*

La animación basada en física es ampliamente utilizada para la creación de personajes o criaturas de cuerpos blandos o *soft-bodies*. Este tipo de criaturas, a diferencia de las hechas de cuerpos rígidos, pueden deformar su cuerpo y tienen una gran aplicación en el ámbito de los video-juegos y películas de animación. Existen muchos trabajos previos sobre técnicas para el control de las formas de los *soft-bodies*. Algunos métodos han propuesto la creación de *keyframes* de una animación usando formas de reposo interpoladas [103], solucionador de Lagrange restringido [18], regulador lineal cuadrático [12], o mediante optimización espacio-temporal [11]. Martín et al. [119] introdujeron un enfoque basado en ejemplos para la simulación de *soft-bodies* con comportamientos deseados. En cuanto al control de la locomoción de *soft-bodies*, ésta ha sido algo menos explorada dentro del campo de la animación. Trabajos previos han demostrado que sistemas de mallas de muelles se pueden utilizar para simular el movimiento de gusanos, serpientes y peces [124], [178], [73]. Miller [124] utilizó fuerzas de fricción de forma que un gusano puede deslizarse hacia adelante mediante la contracción de los segmentos de su cuerpo elástico. Tu y Terzopoulos [178] aplicaron un modelo de dinámica de fluidos para proporcionar empuje hacia adelante cuando un pez deforma su cuerpo. Recientemente, Kim y Pollard [99], [100] demostraron que puede lograrse una locomoción mucho más compleja mediante un simulador de elementos finitos (*FEM*). A pesar de estas y otras técnicas propuestas, en la literatura no existen trabajos relacionados sobre *soft-bodies* en animación construidos mediante enjambres de agentes programables y reconfigurables inspirados en el desarrollo biológico.

## 5.2. Arquitectura modular mediante desarrollo programable en 3D

En esta sección se presenta *MapDevo3D*, un modelo computacional y de simulación de la morfogénesis, programable y reproducible, que combina el auto-ensamblaje mecánico (SA) y la formación de patrones (PF). Estos dos procesos principales son controlados y parametrizados por un genotipo G almacenado en el interior de cada célula del enjambre 3D. Este modelo en 3D es una extensión del modelo original bidimensional [51] para la creación de organismos multicelulares artificiales en enjambre, que se desarrollan y presentan un comportamiento en un mundo virtual físicamente realista. Las propiedades diferenciales de las células (orientación, división, adhesión, movimiento) están determinadas por las regiones de expresión genética a las que pertenecen, mientras que al mismo tiempo, estas regiones se expanden más allá y se segmentan en subregiones debido al auto-ensamblaje de las células diferenciadas (Figura 5.3). Siguiendo las metáforas artísticas empleadas por el biólogo Enrico Coen para describir el desarrollo embrionario [37], se podría decir que PF es similar a un "lienzo que se auto-pinta" y SA es la "auto-conformación de una forma", y su integración mutua crea una escultura auto-coloreada. El presente modelo también puede interpretarse a partir de estas dos diferentes perspectivas: como una especie de movimiento para la formación de patrones en un autómata celular, en el que las células se dividen y espacialmente se reorganizan bajo la influencia de su propio patrón de actividad; o, por el contrario, como una especie heterogénea de movimiento colectivo en un enjambre, en el que las células se diferencian gradualmente y modifican sus interacciones en función de sus posiciones y las regiones que forman.

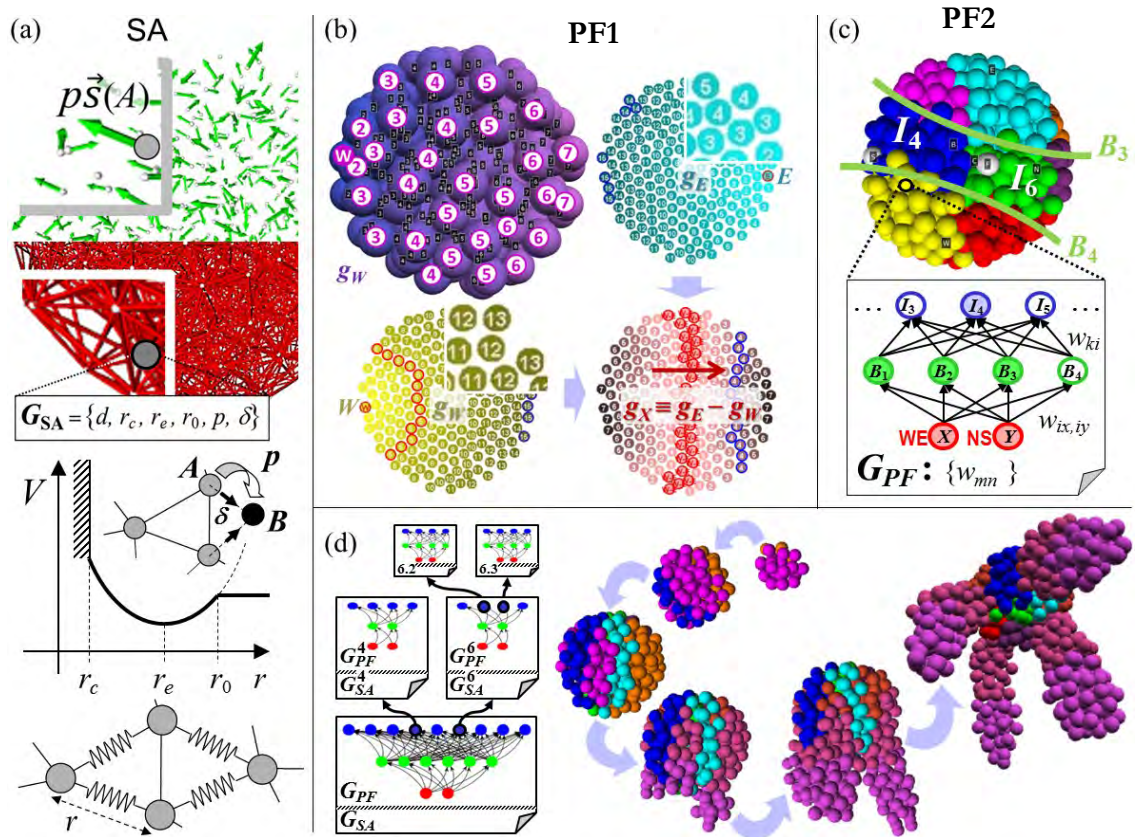


Figura 5.3: Visión general de los mecanismos esenciales de *MapDevo3D*. (a) El autoensamblaje (SA). Abajo: imagen del potencial de adhesión  $V$  entre las células vecinas representado mediante resortes elásticos. En cada paso de tiempo, una célula  $A$  puede dividirse con una probabilidad  $p$ . Arriba, mitad inferior: vista de la malla 3D de interacciones vecinales en un enjambre de 600 células (con recuadro en zoom), cada una con un conjunto de parámetros genéticos  $G_{SA}$ . Arriba, mitad superior: el mismo enjambre que muestra el campo de vectores de división  $\vec{s}$ . (b) La formación de patrones de propagación de información posicional (PF1). Gradientes circulares de "salto de contadores"  $g$  originados a partir de células origen  $W$  (oeste) y  $E$  (este), y coloreados según su valor. Izquierda: gradientes de  $g_W$  en un enjambre 3D y 2D (con recuadro zoom). Derecha: gradiente  $g_E$  en un enjambre 2D, y la línea ecuatorial  $|g_E - g_W| \leq 1$  (anillos de color rojo). (c) La formación de patrones mediante diferenciación programada (PF2): El mapa de color representa hibridación virtual in situ que revela la "geografía oculta" del embrión. Cada región contiene células de un tipo  $I_k$ , cuyo nivel de expresión es una salida de la red de regulación genética  $G_{PF}$  subyacente, que toma como entrada el valor de los contadores de gradiente. (d) El crecimiento modular y la creación de patrones ( $SA^k + PF^k$ ). Visión idealizada de una red reguladora típica de tres niveles dando lugar a un organismo artificial mediante un proceso simultáneo de crecimiento de las extremidades y diferenciación dentro de estas extremidades.

En la descripción del modelo que se realiza a continuación, las divisiones y los movimientos dentro de un enjambre homogéneo de células 3D (SA puro); y la difusión de la señal y diferenciación celular a través del enjambre (PF puro), se introducen por separado. A continuación, estas dos partes se unen para formar los patrones de crecimiento reproducibles (SA + PF). Finalmente, esta combinación se repite en módulos (SA<sup>K</sup> + PF<sup>K</sup>) dentro de un gran sistema heterogéneo para crear morfologías complejas por un proceso de refinamiento recursivo de detalles. La Figura 5.3 muestra una vista general de estos mecanismos que serán presentados a continuación. Para el desarrollo de este modelo 3D se ha utilizado el motor físico ODE usando *Python* como lenguaje de programación.

### 5.2.1. El crecimiento y desarrollo de un enjambre 3D homogéneo (SA)

El modelo consta de un enjambre 3D de pequeñas células esféricas que incorporan dos aspectos principales de la biomecánica celular: la adhesión celular, en forma de reordenamiento elástico, y la división celular (Figura 5.4). Las células permanecen unidas unas a otras mediante afinidades mutuas de adhesión, representadas teóricamente mediante un potencial local de interacción  $V$  (Figura 5.4a) entre pares de vecinos, basado en tres partes: (i) de repulsión para  $r < r_c$ , (ii) atracción cuadrática (elástica) alrededor  $r_c$ , y (iii) potencial plano para  $r > r_0$ . En la práctica, estos enlaces entre células son implementados mediante resortes elásticos cuya dinámica resulta de aplicar estas ecuaciones sobre las células:  $\vec{F}_k = -k\vec{l}$  (ley de Hook), y  $\vec{F}_c = -c\vec{l}$  (fuerza de amortiguación), donde  $\vec{l}$  es el vector de desplazamiento del enlace (es decir, la distancia y la dirección en la que se deforma). Las relaciones de vecindad son calculadas por una triangulación de *Delaunay* en 3D (Figura 5.4b), de la que se eliminan los enlaces de larga distancia por encima de un punto de corte  $r_0$ . A partir de un pequeño grupo de células de partida, las células comienzan a dividirse en cada paso de tiempo  $\Delta t$ , con probabilidad  $p$ , en la dirección de los vectores locales  $\vec{s}$

(Figura 5.4c-d) y a una pequeña distancia inicial  $\delta$ , aumentando tanto el número de células como la disposición espacial del enjambre. Las nuevas células creadas en cada  $\Delta t$  inmediatamente crean los enlaces de su propio vecindario y participan en la dinámica de fuerzas elásticas del enjambre. Las probabilidades de división homogéneas  $p$  y las orientaciones uniformemente aleatorias  $\vec{s}$ , mantienen el enjambre en forma de bola simétrica en todo momento (Figura 5.4e). Sometidas a las fuerzas elásticas las células se reorganizan constantemente en una malla casi regular cuando se alcanza el equilibrio. En resumen, los parámetros globales  $SA$  (valores idénticos en todas las células) constituye el genoma  $G_{SA}$ , mientras que los parámetros y variables (valores específicos para cada célula  $A$ ) son almacenados en registros denotados  $Q_{SA}(A)$ , el cuál determina el estado local interno de la célula  $A$ . En esta primera fase del desarrollo, caracterizada por un crecimiento simétrico del cuerpo, estos conjuntos vienen definidos por:

$$G_{SA} = \{r_c, r_e, r_0, p, \delta\}$$

$$G_{SA}(A) = \{r(A), s(A), N(A), \mathfrak{R}(A)\},$$

donde  $\mathfrak{R}(A) = \{\vec{r}(A, B); B \in N(A)\}$  es el conjunto de vectores de los enlaces conectados a  $A$ , es decir, contiene toda la información espacial del vecindario de  $A$ . Destacar que, en posteriores fases del desarrollo, ciertos aspectos del modelo podrán ser representados tanto por las parámetros genéticos globales como por los valores de estado local.

### 5.2.2. La propagación de información posicional en un enjambre 3D (PF1)

Las piezas de un rompecabezas también se definen por la imagen que llevan. En la auto-formación del enjambre, este papel es desempeñado por las variables de estado que determinan la actividad  $PF$  dentro de cada célula. El modelo distingue entre dos

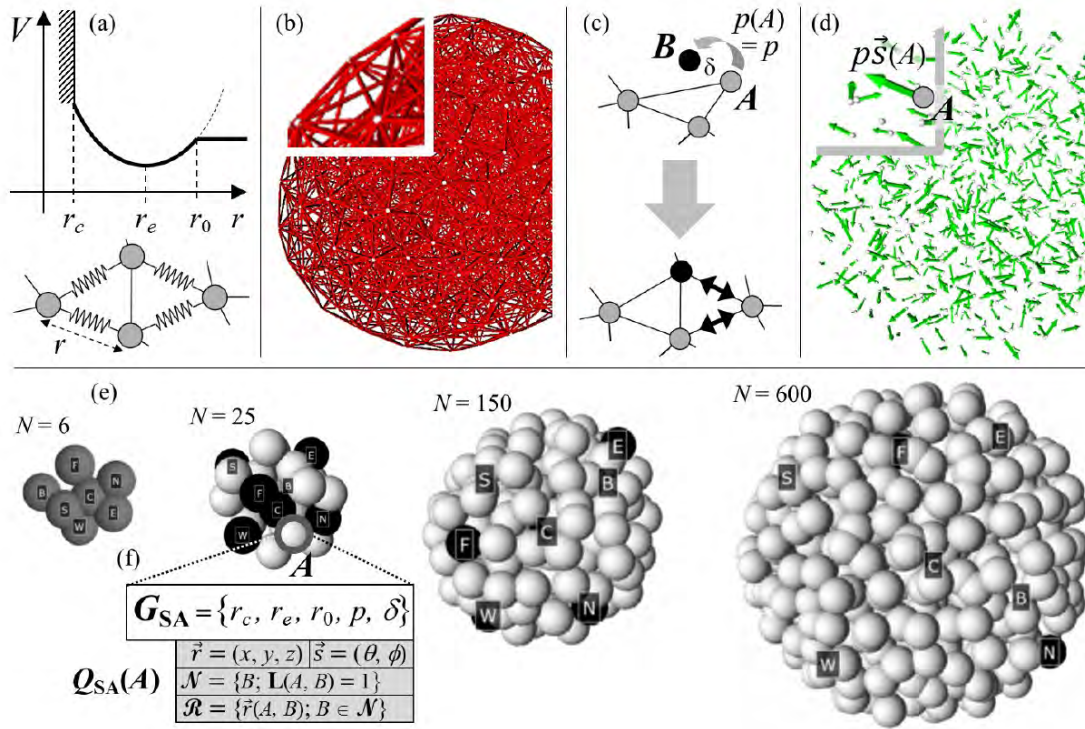


Figura 5.4: Crecimiento de un enjambre 3D homogéneo (fase de auto-ensamblaje, SA). (a) Gráfica del potencial de adhesión  $V$  entre las células vecinas, representado mediante resortes elásticos. (b) Visión completa de la malla 3D de triangulación de *Delaunay*, podada a distancia de corte  $r_0$ , en un enjambre de 600 células obtenida por simulación (ampliación para más detalle de la esquina superior izquierda). Los enlaces de color rojo (resortes elásticos) representan las interacciones entre las células vecinas. (c) En cada paso de tiempo, una célula  $A$  se puede dividir con probabilidad local  $p(A)$  y crear a una pequeña distancia una nueva célula  $B$  que inmediatamente participa en la dinámica de fuerzas elásticas. (d) Vista del mismo enjambre que muestra el campo de vectores de división. Cada flecha verde representa  $p(A)\vec{s}(A)$ , donde  $\vec{s}(A)$  es el vector unitario perpendicular al plano de escisión de  $A$  (aquí, con una orientación aleatoria). (e) La simulación del crecimiento de un enjambre, tanto en número de células  $N$  como en extensión espacial. Las probabilidades de división homogéneas y orientaciones uniformemente aleatorias mantienen el enjambre en forma de bola simétrica en todo momento y escala (etiquetas con letras en ciertas células emisoras de gradiente, explicado más adelante). (f) La representación simplificada del conjunto idéntico de parámetros genéticos  $SA$  dentro de cada célula, denotado por  $G_{SA}$ , y los registros del estado interno  $SA$  en la célula  $A$ , denotado por  $Q_{SA}(A)$ .



tipos de variables de estado  $PF$  específicas: variables gradiente ( $PF1$ ) y las variables de expresión ( $PF2$ ). En primer lugar, las variables de señalización de gradiente se propagan y establecen la información posicional a través del enjambre (Figura 5.5) y son almacenadas en el conjunto de registros  $Q_{PF1}(\mathcal{A})$ . En sistemas artificiales como la computación amorfa o la computación espacial, la idea de información posicional basada en gradientes es comúnmente implementada mediante contadores de saltos (*hop-count*) en cada agente. El modelo presentado aquí también implementa este mecanismo, almacenando cada célula seis contadores  $g_W, g_E, g_N, g_S, g_F, g_B \in \{0,1,2,\dots\}$ , que varían de forma regular e incremental a lo largo del enjambre. Por ejemplo, una célula origen  $W$  (oeste) contiene al inicio su  $g_W = 0$ , que pasa con valor 1 a sus vecinos, que a su vez le indican a sus vecinos que establezcan  $g_W$  a 2, y así sucesivamente. Este mecanismo se rige bajo tres reglas: (1), cada célula  $\mathcal{A}$ , envía a todos sus vecinos el valor de su contador de gradiente incrementado en uno,  $g_W(\mathcal{A})+1$ . (2) Para dar a esta propagación isotrópica una dirección específica, una segunda regla local instruye a cada célula para que conserve sólo el valor más pequeño de entre su valor actual y los valores recibidos por sus vecinos. (3) Por último, para crear un gradiente monótonamente creciente, una tercera regla local requiere a cada célula que compruebe regularmente que al menos uno de sus vecinos tiene un valor inferior al suyo propio y si no, entonces debería “vaciar” su actual valor estableciéndola en  $+\infty$ , es decir, estará listo para aceptar cualquier nuevo valor en el siguiente paso. El resultado de la aplicación recurrente de estas reglas es la propagación de una onda más o menos circular de los valores  $g_W$  a través del enjambre y centrada en  $W$  (Figura 5.5a). De la misma manera, en el lugar más alejado de  $W$ , una célula origen  $E$  (este) crea el mismo tipo de gradiente en la dirección opuesta. Juntos forman un gradiente derivado  $WE$  que genera una especie de eje horizontal  $X$ , cuyos valores provienen de calcular la diferencia polar  $g_X = |g_E - g_W|$ . Junto con otros dos pares de células origen de gradiente, ( $N$  (norte),  $S$  (sur)) ( $NS$ ) y ( $F$  (frontal),  $B$  (trasera)) ( $FB$ ), forman un

sistema de coordenadas 3D ( $g_x, g_y, g_z$ ) basado en secciones ecuatoriales (conjunto de células donde los contadores opuestos son iguales  $\pm 1$  (Figura 5.5c)). Un ejemplo de cómo se pueden usar los gradientes se muestra en la Figura 5.5e, adelantando en parte el mecanismo de diferenciación *PF2* que se explica en la siguiente sección. Aquí, las zonas centrales horizontales y verticales (representadas en color rojo) son determinadas por  $|g_x(A)| < WE/3$  y  $|g_y(A)| < NS/3$ , de forma que si una de las dos condiciones es cierta, entonces la célula  $A$  cambia su color a rojo. Lo interesante es que las células origen de gradientes se auto-posicionan continuamente alejándose una de la otra, maximizando sus contadores internos como se explica a continuación. En el caso del par  $WE$ , cada célula contiene una variable bandera booleana,  $\gamma_W$ , que es verdadera sólo en la célula  $W$  y falsa en el resto. En cada paso de tiempo  $\Delta t$ ,  $W$  comprueba el valor de  $g_E$  de todos sus vecinos y en el caso de tener un valor más alto este vecino pasaría a ser el nuevo  $W$ . Lo mismo ocurre en el caso de  $E$  y, por extensión, en los pares  $NS$  y  $FB$ . Usando gradientes, la proliferación o división celular en  $SA$  también se puede regular con un parámetro umbral  $g_{max}$ : cuando uno de sus contadores de gradiente  $g_W, g_E, \dots, g_B$  alcanza  $g_{max}$ , las células dejan de dividirse y envían una señal de parada a sus vecinos, por lo que toda la región finalmente se asienta en un tamaño fijo.

En resumen, todos los contadores de gradiente (elementales y derivados), banderas, y tamaño global son almacenados en  $G_{PF1}(A)$ , el cual constituye el estado interno de gradiente de cada célula  $A$  (Figura 5.5f). Por otro lado, parámetros genéticos globales no se necesitan, por lo tanto quedaría:

$$G_{PF1} = \phi$$

$$G_{PF1}(A) = \{g_W, g_E, g_N, g_S, g_F, g_B, \gamma_W, \gamma_E, \gamma_N, \gamma_S, \gamma_F, \gamma_B, WE, NS, TB\}$$

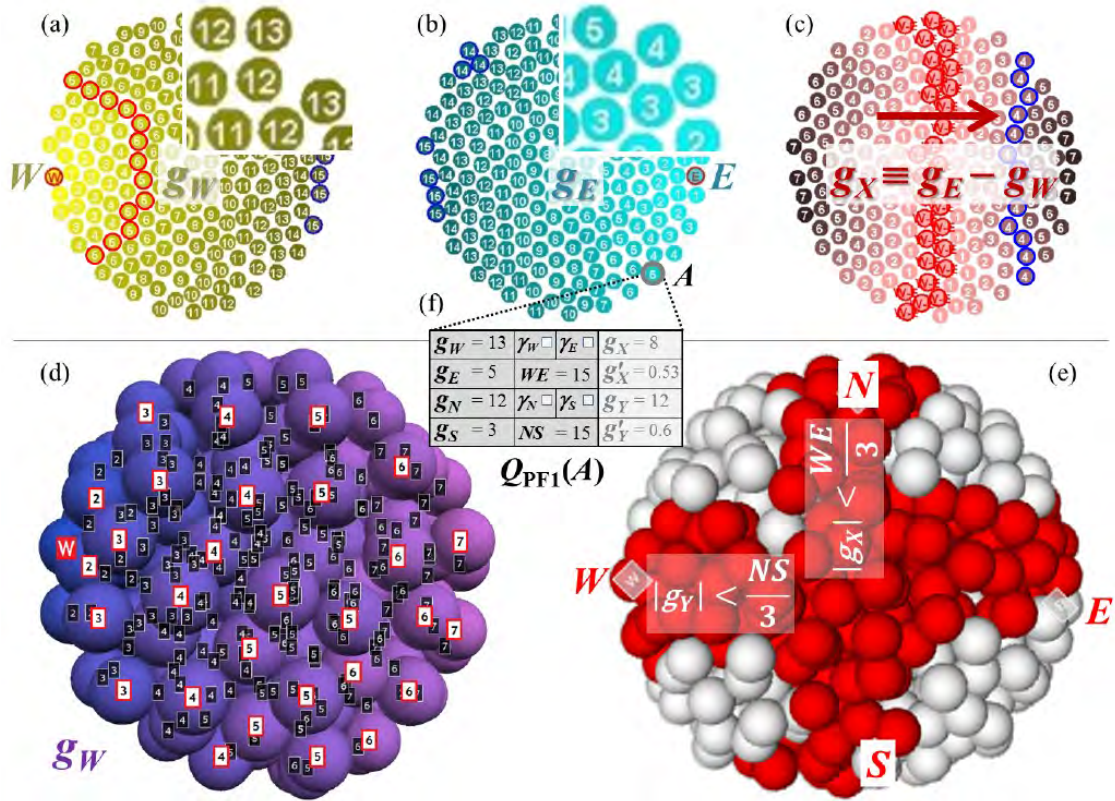


Figura 5.5: Propagación auto-organizada de la información posicional (*PFI*). Varias vistas de un enjambre 2D (para una visión simplificada) y 3D obtenidos por simulación y etiquetadas. (a) Gradiente radial en un enjambre en forma de disco 2D visualizado mediante un mapa de color amarillo-sombreado (ampliación en la esquina superior derecha). (b) Gradiente contrario  $g_E$  procedente de la célula origen  $E$  en el mismo enjambre 2D. (c) Valor del gradiente secundario  $g_X$  derivado de los dos anteriores estableciendo  $g_X = |g_E - g_W|$ . (d) Gradiente esférico 3D de los contadores  $g_W$  originados por la célula origen  $W$ . Se muestran todos los valores de gradiente (etiquetas negras, incluyendo las células ocultas por la perspectiva) y una estimación del gradiente general (etiquetas blancas en cuadros rojos). El mapa de color de las células varía linealmente con  $g_W$  de azul a púrpura. (e) Ejemplo de gradiente creando un dominio en forma de cruz en el centro del enjambre de células. (f) Registro  $Q_{PFI}(A)$  (para un enjambre 2D) que almacena el estado de gradiente interno en cada célula  $A$ . No hay parámetros genéticos globales para los gradientes, por lo tanto  $G_{FP1} = \phi$ .

### 5.2.3. Diferenciación programada en un enjambre 3D (PF2)

Tras la propagación auto-organizada de la información posicional, los valores de los patrones se calculan a partir de los gradientes estableciéndose la diferenciación celular. Este proceso marca el surgimiento de la heterogeneidad, es decir, la segmentación del enjambre en regiones de identidad o tipos celulares. Para este objetivo, cada célula implementa una abstracción de una red de regulación genética simple (GRN) (Figura 5.6), denotada  $G_{PF2}$ , cuyos pesos representan los parámetros genéticos del proceso PF2. La GRN que se utiliza en este modelo es una *feed-forward* de tres capas que imita de una forma muy sencilla y abstracta las dinámicas de las regulaciones reales, ya que no contiene enlaces recurrentes. Sin embargo, es muy similar a la inicial de cinco niveles en *Drosophila*. Cada región de identidad  $\{I_k\}_{k=1\dots m}$ , donde  $I_k \in [0,1]$ , refleja a alto nivel la expresión de un gen de identidad particular:  $I_1, I_2, \dots, I_m$  (capa de salida). Estos genes de salida son una función de la entrada formada por las coordenadas  $g_X, g_Y, g_Z$  (capa de entrada) derivadas del valor de los gradientes  $g_W, g_E, \dots, g_B$ , a través de la expresión de genes intermedios de segmentación  $\{B_i\}_{i=1\dots n}$ , donde  $B_i \in [0,1]$ , los cuales dividen el enjambre en diferentes partes (capa intermedia). En la práctica, los nodos  $B_i$  se comportan como funciones lógicas, y los pesos  $w_{ix,ij,iz}$  de la red determinan dónde se establecen los límites mediante los cuales se divide el enjambre en regiones, de acuerdo a  $g_X, g_Y, g_Z$ . Los nodos  $I_k$  reciben a su vez las salidas  $[0,1]$  provenientes de la capa intermedia y nuevamente mediante otra función lógica determinan el tipo celular en función de los pesos  $w_{ki}$  determinados en  $G_{PF2}$ . En resumen, los parámetros genéticos globales  $G_{PF2}$  determinan el funcionamiento general de la GRN que establece las regiones de expresión genética, mientras que los parámetros y variables locales de cada célula  $A$  (en este caso  $g_X, g_Y, g_Z$ ) son almacenados en registros denotados  $G_{PF2}(A)$ , los cuales representan la entrada particular de cada célula en la GRN. Estos conjuntos vienen definidos por:

$$G_{PF2} = \{\{I_k\}_{k=1\dots m}, \{B_i\}_{i=1\dots n}, w_{ix,ij,iz}, w_{ki}\}$$

$$G_{PF2}(A) = \{g_X, g_Y, g_Z\}$$

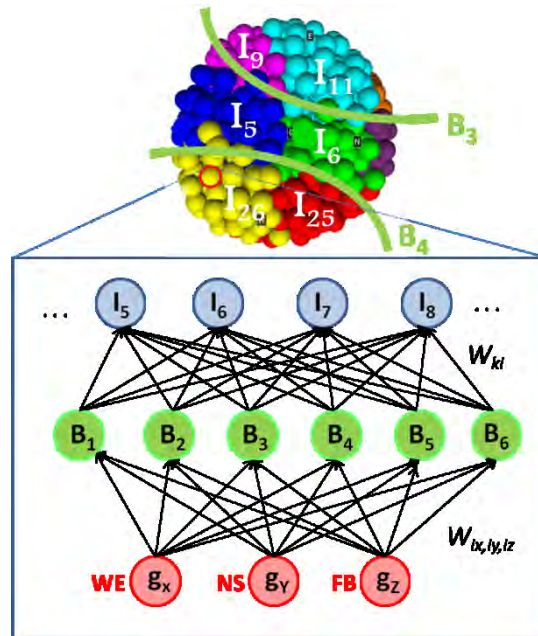


Figura 5.6: Formación de patrones mediante diferenciación programada (*PF2*): El mapa de color representa hibridación virtual in situ que revela la "geografía oculta" del embrión. Cada región contiene células de un tipo  $I_k$ , cuyo nivel de expresión es una salida de la red de regulación genética  $G_{PF2}$  subyacente, que toma como entrada el valor de los contadores de gradiente. En este caso el enjambre de células se segmenta en 27 regiones ( $3 \times 3 \times 3$ ). La capa intermedia contiene tres pares de genes de segmentación que dividen el enjambre en tres regiones cada una, respectivamente, a lo largo de los ejes  $WE$ ,  $NS$  y  $FB$  (éste último eje no se diferencia por la perspectiva de la figura).

#### 5.2.4. Crecimiento y creación de patrones simultáneos ( $SA + PF$ ) en un enjambre 3D

El auto-ensamblaje de un enjambre sin patrones,  $SA$ , y la creación de patrones de un enjambre dado,  $PF$ , se combinan para crear patrones de crecimiento. Los agentes ajustan continuamente sus posiciones de acuerdo con las restricciones elásticas  $SA$ , al mismo tiempo que continuamente intercambian valores de gradiente y las señales de  $PF$  sobre los enlaces dinámicos. La dinámica dual  $SA + PF$  se guía por un genotipo combinado  $G = G_{SA} \cup G_{PF}$ . Durante la división celular, cualquier célula  $B$

generada por una célula  $A$  hereda todos los atributos de  $A$ , incluidas  $G$  y sus variables de estado internas. Inmediatamente comienza contribuyendo a las fuerzas  $SA$  y al tráfico de los gradientes de  $PF$  que mantienen la consistencia del patrón en todo momento en el enjambre. La Figura 5.7 muestra diferentes etapas de crecimiento del enjambre en el que se puede apreciar la actuación simultánea de  $SA + PF$ .

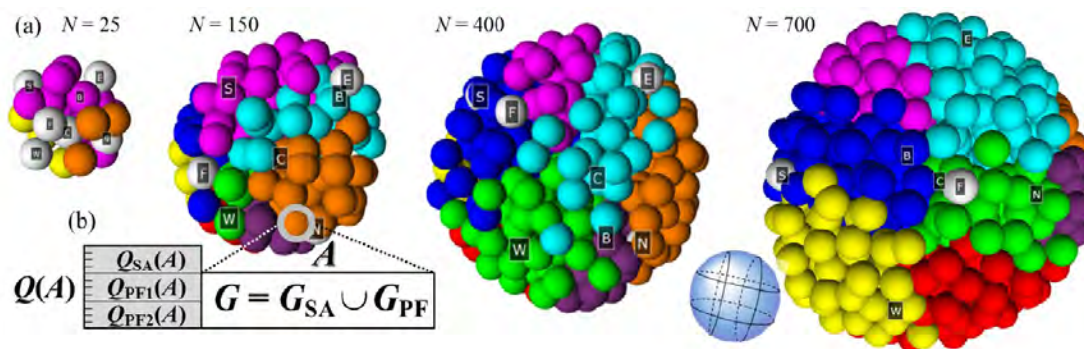


Figura 5.7: Crecimiento y patrón simultáneos ( $SA + PF$ ). (a) Un enjambre 3D en desarrollo para el tamaño  $N = 700$ . Las células se dividen y se empujan entre sí de acuerdo a las dinámicas  $SA$ . Al mismo tiempo se llevan a cabo los procesos  $PF1$  (información posicional) y  $PF2$  (diferenciación). (b) El genoma combinado dentro de cada célula es la unión de los genomas de auto-ensamblaje y la formación de patrones vistos anteriormente.

### 5.2.5. Patrón recursivo modular ( $PF^K$ ) y crecimiento anisotrópico ( $SA^K$ ) en un enjambre 3D

Los embriones no se desarrollan en una sola fase, sino que lo hacen en numerosas etapas incrementales. Para seguir el ejemplo de *Drosophila*, las regiones adquieren primero las identidades de pierna, ala o antena a través de la difusión global, para a continuación, desarrollar los sistemas de coordenadas locales de gradientes de morfógenos para formar el miembro u órgano previsto. Para reflejar esto, la red genética  $G_{PF}$  se extiende para incluir una jerarquía de módulos de red que pueden

generar patrones de una manera recursiva (Figura 5.8a). En primer lugar, la red de base  $G_{PF}$  establece regiones principales de identidad como antes. Entonces, unas subredes  $G_{PF}^k$  disparadas por nodos  $I_k$  en  $G_{PF}$ , particionan aún más estas regiones en compartimentos de identidad más pequeños y en una escala más fina. La modularidad, un principio que la evolución biológica “descubrió” naturalmente, es también deseable en arquitecturas robóticas o de software.

Por otra parte, para obtener verdaderas dinámicas de deformación, y conferir formas no triviales al sistema más allá de simples globos o bolas, las células deben ser capaces de diversificar sus características  $SA$  en función de su tipo  $PF$  y su posición espacial, cerrando así el ciclo de retroalimentación entre la  $SA$  y  $PF$ . En particular, tienen que exhibir una división celular no homogénea, anisótropa (variando  $p$ ) y adhesión diferencial (variando  $V$ ). Por ejemplo, el crecimiento de la estructura de una extremidad (Figura 5.9) se puede lograr mediante una gruesa imitación del crecimiento de las ramas de las plantas. En este proceso, sólo la punta o “meristemo apical” de la extremidad (valores más altos del gradiente  $S$ , entendiendo que la célula  $S$  esté situada en la parte de la extremidad más cercana al cuerpo) se dividen activamente en cualquier momento. Por otra parte, los parámetros de  $V$  se pueden programar de tal manera que son atractivos sólo entre células homotípicas (dentro de la extremidad), y repulsivos entre las células heterotípicas. Al igual que la división no homogénea, la adhesión diferencial es una condición esencial de la formación de una forma compleja.

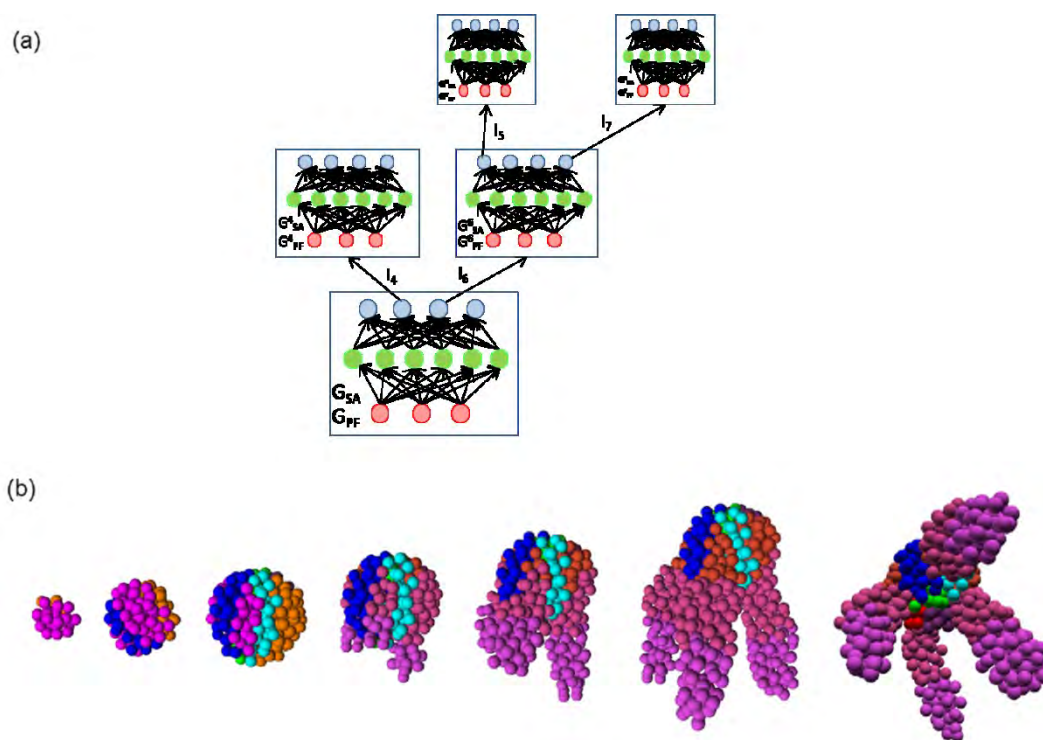


Figura 5.8: Crecimiento modular y la creación de patrones ( $SA^k + PF^k$ ) (a) Visión idealizada de una red reguladora típica de tres niveles dando lugar a (b) un organismo artificial mediante un proceso simultáneo de crecimiento de las extremidades y diferenciación dentro de estas extremidades.

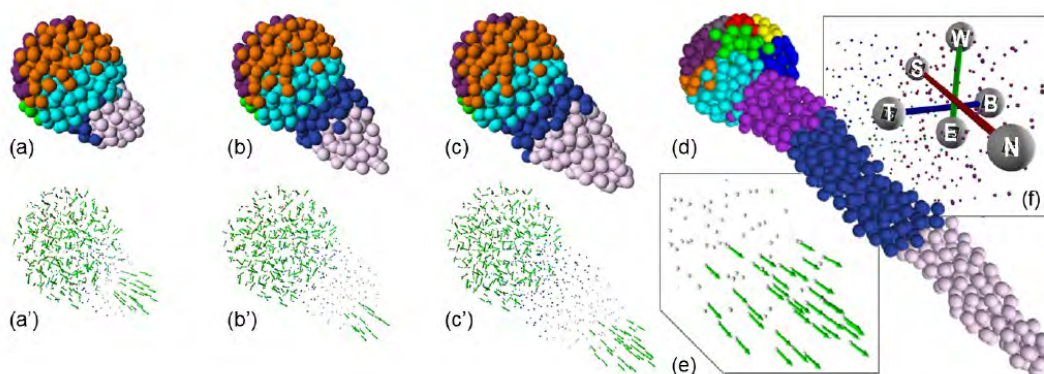


Figura 5.9: Desarrollo de una extremidad individual. (a-d) estados sucesivos. (a'-c') correspondiente campo de división: el cuerpo ha detenido su crecimiento, mientras que en la extremidad la probabilidad división es cero en todas las partes excepto en el extremo situado al Norte ( $g_N \leq 3$ ), estando la célula S (sur) situada en la zona de la extremidad más cercana al cuerpo. (e) Detalle de la división en la punta de la extremidad. (f) Los tres pares de células origen de gradientes auto-posicionadas dentro de la extremidad (mostrando ejes virtuales, no vínculos reales), con la célula Sur en la raíz y el Norte en la punta.



Finalmente, juntando todo, morfologías completas tienen la capacidad de desarrollarse y auto-organizarse a partir de unas pocas células (Figura 5.8b). Estas morfologías son complejas arquitectónicamente, ya que pueden estar hechas de cualquier número de diversos módulos y partes que no necesariamente se repiten en formas periódicas o triviales. Son fenotipos programables que salen del mismo genotipo expresado por cada célula del enjambre. También son reproducibles, ya que sus estructuras morfológicas no se dejan al azar, sino que son controladas por el genotipo. Las posiciones exactas de las células a nivel microscópico son al azar, pero no las regiones macroscópicas que forman. La modularidad del fenotipo es un reflejo directo de la modularidad del genotipo: las dinámicas jerárquicas  $SA + PF$  se desarrollan de forma recursiva dentro de las diferentes regiones y subregiones que se crean. Cada bloque  $SA^k + PF^k$  puede tener diferentes parámetros genéticos internos  $SA$  y  $PF$ , potencialmente dando a cada región un comportamiento morfodinámico y una actividad diferentes. La integración entre  $SA$  y  $PF$  sucede en los nodos de identidad  $I_k$ , de igual forma que estos nodos activan la actividad de expresión genética en módulos subordinados  $G_{PF}^k$  para crear nuevos patrones de segmentación locales, simultáneamente activan los cambios de comportamiento en los módulos subordinados  $G_{SA}^k$  para crear nuevos comportamientos morfodinámicos a la misma escala.

### 5.3. Comportamiento de las arquitecturas funcionales en un entorno físico

Aunque ya supone un reto en sí mismo la tarea de diseño de las leyes del metadiseño del desarrollo artificial inspirado en la biología, sólo constituye la primera parte del esfuerzo de la ingeniería embriomórfica. ¿Qué capacidades de recepción-actuación y de comportamiento puede llevar a cabo un organismo generado? ¿Qué representan realmente sus células (agentes) y los órganos (regiones) y qué pueden lograr en la

práctica?. En los organismos biológicos, aunque la fisiología celular a menudo participa en el desarrollo (por ejemplo, las señales eléctricas de las neuronas que guían la sinaptogénesis), parece que hay una amplia distinción entre los genes de desarrollo y el resto del genoma. En los sistemas computacionales, estos dos modos también podrían ser desacoplados en dos conjuntos diferentes de variables de estado. Después de alcanzar la maduración del desarrollo, y mientras todavía conserva las tareas de mantenimiento y auto-reparación, la actividad morfogénica  $SA + PF$  daría paso a otro tipo de actividad relacionada con el cómputo funcional.

En esta sección se describe la transición de objetivos morfológicos a funcionales a través de organismos *MapDevo3D* animados inmersos en un entorno virtual. Después de que el organismo se ha desarrollado plenamente a través de los procesos descritos en la sección 5.2, debe interactuar con un mundo exterior sencillo, formado por un suelo rígido y posibles obstáculos en un campo gravitacional (simulado aquí con el motor de física *ODE*). Para exhibir movimiento, locomoción y comportamiento primitivo, los organismos contraen los enlaces de adhesión entre células “musculares”, mientras que otras células se diferencian como “huesos” y “articulaciones” para soportar y articular la estructura del cuerpo (Figura 5.10). Por último, una exploración paramétrica y búsqueda evolutiva introducida en la sección 5.4 completan esta demostración original de un sistema de *Alife evo-devo*, en el que la auto-organización no sólo es programable, sino también funcional y capaz de evolucionar.

### 5.3.1. Diferenciación mecánica en huesos y músculos

Dentro del paradigma embriomórfico, el desarrollo guiado por genotipo de un organismo no sólo proporciona una forma general reproducible, sino también puede equipar a esta forma con características estructurales incorporadas que le confieren propiedades mecánicas específicas. En la Figura 5.10, por ejemplo, unas pocas células en la base de las extremidades se han diferenciado como “músculos”,

mientras que otras se han convertido en “huesos” dentro de las extremidades, y “articulaciones” en la unión entre las extremidades y el cuerpo. Computacionalmente, esto equivale a la adición de varios campos booleanos (funciones de los gradientes locales) a cada módulo genético (Figura 5.10b). Aquí, el campo muscular corresponde a la sección base de una extremidad, por ejemplo, donde  $g_S \leq 5$  (región de color rosa, diferente de la región morada que representa la parte más externa de la extremidad). Mientras que el campo de hueso se extiende sólo a lo largo de una línea delgada desde el Sur al Norte en cada extremidad ( $|g_W - g_E| \leq 1$ ) y que se encuentra aproximadamente en el centro de la extremidad (como los huesos reales en un cuerpo biológico). Los tipos de enlace se deducen a continuación mediante la conexión de las células vecinas de tipos idénticos: por ejemplo, los enlaces de hueso (en color blanco) se forman exclusivamente entre las células óseas. En este caso, que un enlace se convierta en “hueso” significa hacerse rígido, es decir, la adquisición de un coeficiente de resorte virtualmente infinito, de manera que se mantiene una relación espacial fija entre las dos células unidas por ese enlace. El efecto es que una estructura ósea conectada forma un esqueleto que puede soportar todo el organismo y mantenerlo erguido sobre el suelo a pesar de la atracción gravitatoria (Figura 5.10e-g).

#### 5.3.2. Animación mediante contracción muscular

Por último y lo más importante, una vez que las características mecánicas de las células y los enlaces se han establecido con el desarrollo, el organismo es inmerso en un entorno físico donde puede presentar locomoción y otros tipos de comportamiento. En la figura 5.10c-d, se muestra un organismo caminando en el suelo y pateando una pelota. Esto se logra esencialmente permitiendo que regiones musculares específicas (bases de color rosa de las extremidades) se contraigan periódicamente y de manera no uniforme de acuerdo con un esquema o programa

de coordinación y control predeterminado que debería ser la tarea de un futuro sistema nervioso central. Esta contracción muscular para cambiar la forma del cuerpo y provocar movimiento está en la línea de la *soft-robotics* y de la animación de personajes o criaturas *soft-bodied*.

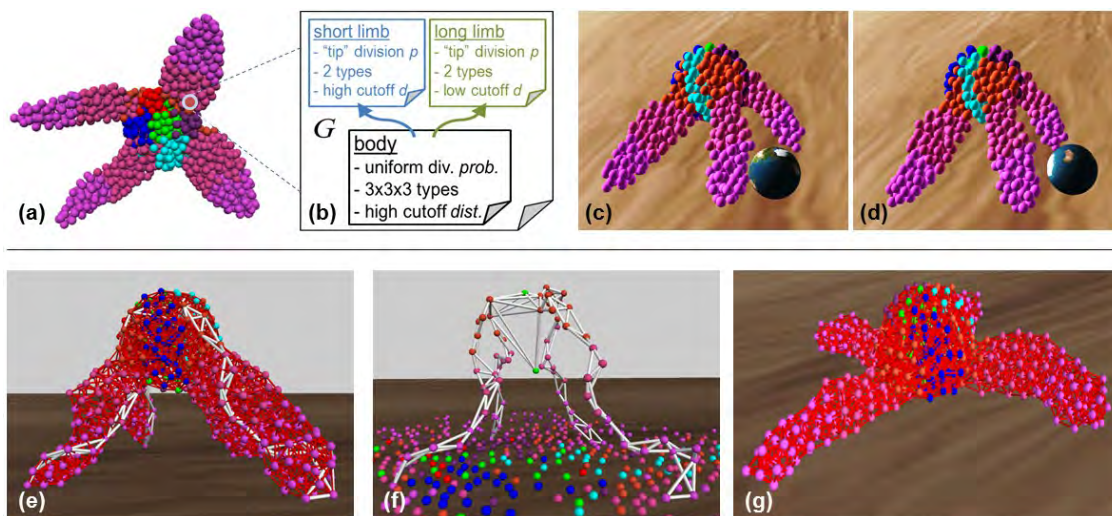


Figura 5.10: Dinámicas y diferenciación estructural. (a) Organismo que ha crecido por completo. Dispone de dos extremidades frontales largas y delgadas y dos extremidades traseras cortas y gruesas (b) programa genético  $G$  ejecutado por todas las células durante el desarrollo, que comprende tres módulos: un módulo cuerpo (probabilidad de división uniforme, 27 tipos de células), un módulo de extremidad corta (división de la región en dos subtipos), y un módulo de extremidad larga. Cada módulo de extremidad se activa en dos regiones diferentes del cuerpo, creando un total de cuatro extremidades. (c-d) Comportamientos de locomoción y golpeo de pelota se consiguen mediante la extensión y la contracción de las regiones “músculo” (subregión de color rosa de las extremidades) en subregiones específicas a intervalos de tiempo específicos, un programa de coordinación y control hace la tarea de un sistema nervioso central. (e) El organismo adulto también contiene un esqueleto hecho de células diferenciadas como “huesos” y enlaces rígidos que las conectan (se muestran en color blanco). (f) Experimento donde se han eliminado todos los enlaces elásticos (color rojo), mostrándose la estabilidad de la estructura esqueleto bajo la atracción gravitatoria. (g) El experimento opuesto donde se eliminan los huesos: el organismo se extiende en el suelo como una estrella de mar debido a la fuerza gravitatoria, ya que pierde la estructura que lo sustentaba.

## 5.4. Función a partir de la estructura, estructura a partir del desarrollo: Un proceso evolutivo doblemente indirecto

*MapDevo3D* propone principios para el metadiseño de organismos robotizados auto-organizados capaces de crear morfologías precisas de una manera puramente endógena. Establece normas genéricas para la emergencia de estructuras programables no aleatorias que no son ni repetitivas ni impuestas por las condiciones externas. Más allá de los estereotipos ingenieriles de mapeos genotipo-fenotipo, el crecimiento también debe ser adaptativo. Es importante ser capaz de diseñar estructuras dinámicas que se pueden desarrollar en una situación que cambia rápidamente, permaneciendo abiertas a las influencias y modificaciones que vienen del entorno en el que se espera que funcione (Figura 5.11). Esto podría ocurrir en múltiples niveles taxonómicos: en una escala de tiempo a largo plazo a través de especiación reflejando nuevos genotipos (Figura 5.11d), o en una escala de tiempo más a corto plazo a través de polimorfismo de una sola especie (Figura 5.11c), o incluso en una escala de tiempo a nivel de individuo mediante polifenismo desarrollado (Figura 5.11b).

### 5.4.1. Polimorfismo Evolutivo: variando el genotipo

Un genotipo  $G$  puede proporcionar parámetros internos para controlar diferentes rasgos de la estructura final: ligeras variantes en el genotipo pueden producir ligeras variantes en el fenotipo (Figura 5.11c). Esto es similar a las leyes clásicas de la genética en poblaciones de la misma especie, lo que corresponde esquemáticamente a los conceptos de alelos o polimorfismos de un solo nucleótido en el ADN. Variando y combinando parámetros genotípicos se da lugar a una familia de diferentes razas; como con los guisantes de Mendel o pichones de Darwin. Hay que tener en cuenta, sin embargo, que la distinción entre el polimorfismo y la

especiación (Figura 5.11d) no es clara: es sólo una cuestión de grado y tiempo, ya que los mismos mecanismos evolutivos están trabajando en ambos casos.

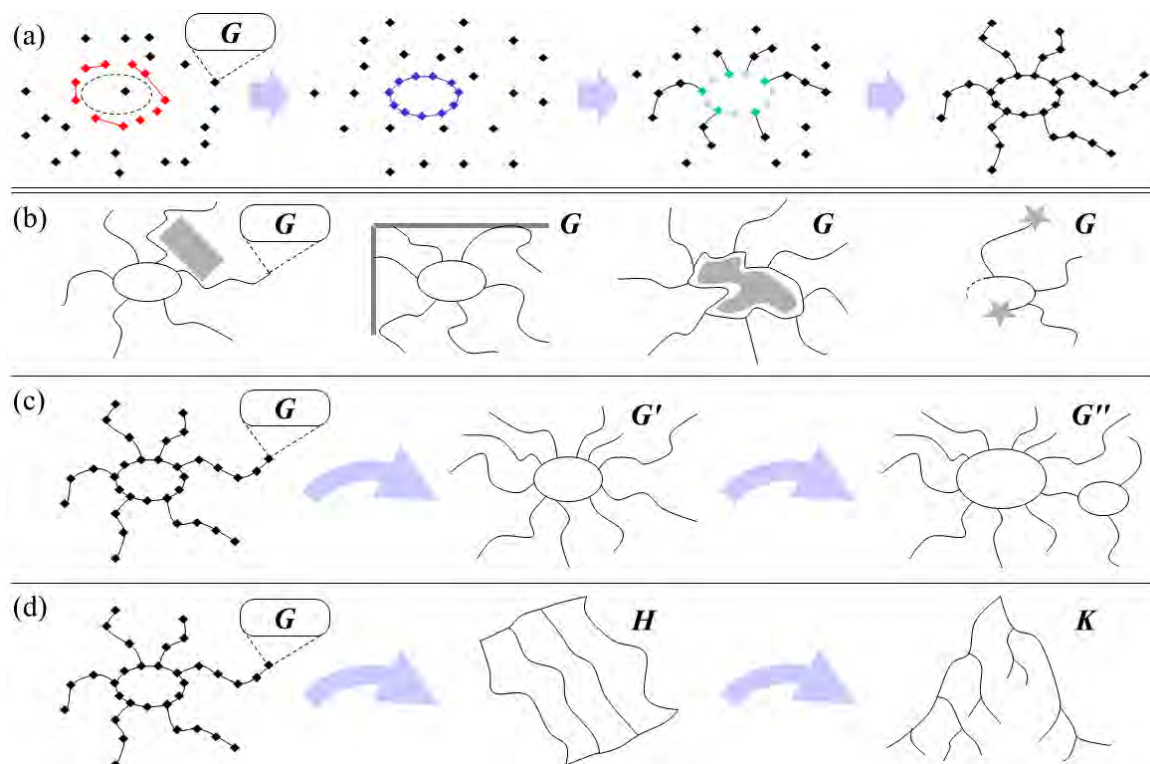


Figura 5.11: Ilustración de varios tipos de adaptación fenotípica en un modelo de crecimiento programable. (a) Desarrollo estereotipado: un cierto genotipo  $G$  da a unos nodos unas instrucciones hacia el auto-montaje en una cierta forma, en este caso una formación esquemática de araña hecha por un anillo y seis extremidades. (b) Polifenismo desarrollado: similar a una planta, el mismo  $G$  da lugar a variantes de la forma anterior modificada por las condiciones externas del medio ambiente, tales como obstáculos o atractores. (c) Polimorfismo: variantes paramétricas leves de  $G$ , denotadas por  $G'$  y  $G''$ , producen otras variantes estructurales, como el tamaño del anillo, número de extremidades, o la ubicación del anillo. (d) Especiación: genomas drásticamente diferentes, denotados como  $H$  y  $K$ , crean diferentes estructuras, aunque no hay diferencia cualitativa real con el caso anterior, ya que sólo es una cuestión de grado y escala de tiempo de la evolución.

5.4. Función a partir de la estructura, estructura a partir del desarrollo: Un proceso evolutivo doblemente indirecto

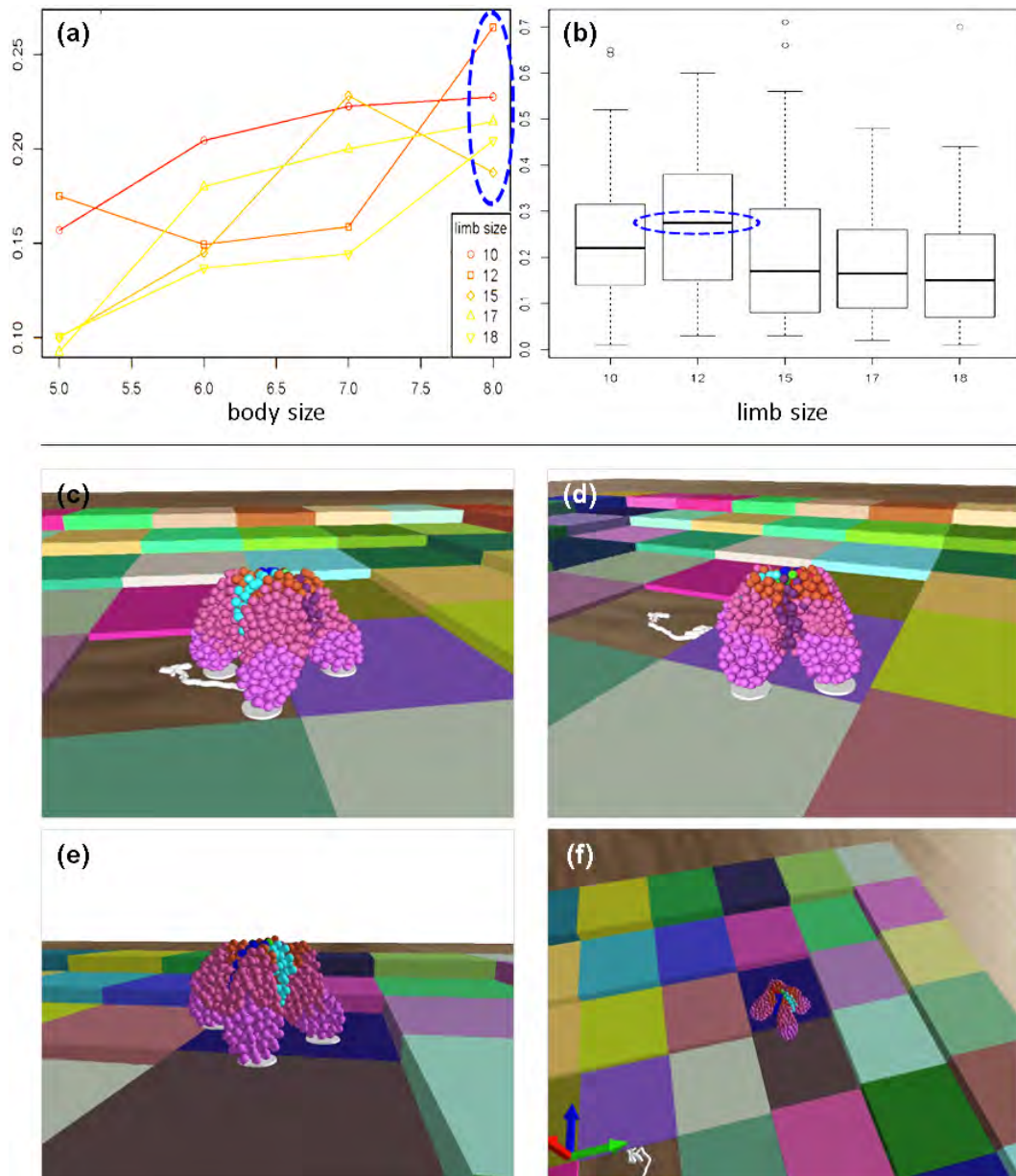


Figura 5.12: Exploración paramétrica simple de un organismo *MapDevo3D* en una tarea de subir escaleras. La función de bondad mide la rectitud de la trayectoria seguida (distancia directa dividida por la trayectoria real, siendo siempre este valor  $< 1$ ). El genoma básico  $G$  determina un organismo de cuatro extremidades, de las cuales dos parámetros varían: el tamaño del cuerpo, representada por un valor máximo de gradiente  $g^b_{max}$ , que detiene la división celular en el cuerpo, y el tamaño de la extremidad, representada por un parámetro similar  $g^l_{max}$ . (a) El aumento de tamaño del cuerpo ante diferentes tamaños de las extremidades: valores medios calculados del valor de bondad de más de 16 individuos para cada par de tamaños cuerpo-extremidad parecen indicar que las criaturas de mayor tamaño funcionan mejor. (b) El aumento de tamaño de las extremidades para diversos tamaños del cuerpo: los valores muestran un ajuste óptimo de las extremidades en  $g^l_{max} = 12$ . (c-f) Cuatro instantáneas del mejor organismo subiendo escaleras. En blanco: traza dejada por el centro de masa de la criatura. Para evitar que la criatura se balancee demasiado alrededor del mismo lugar, debido a la alta granularidad de sus pies, fue dotada con una especie de herraduras (discos en color gris).

### 5.4.2. Polifenismo desarrollado: variando el fenotipo

Bajo un genotipo invariable, sin embargo, el desarrollo puede también ser modificado por las condiciones ambientales (Figura 5.11b). Las señales externas que rodean a un individuo durante su crecimiento también pueden desempeñar un papel importante en su estructura final. Este es el nivel del fenotipo, por el cual analogías naturales se pueden encontrar con mayor facilidad en el reino vegetal, a diferencia de los animales: las plantas y los árboles pueden ser podados, curvados, o esculpidos, ya sea por intervención humana (bonsáis, espalderas, etc.) o por las condiciones naturales (viento, las rocas, el suelo, la luz, etc.).

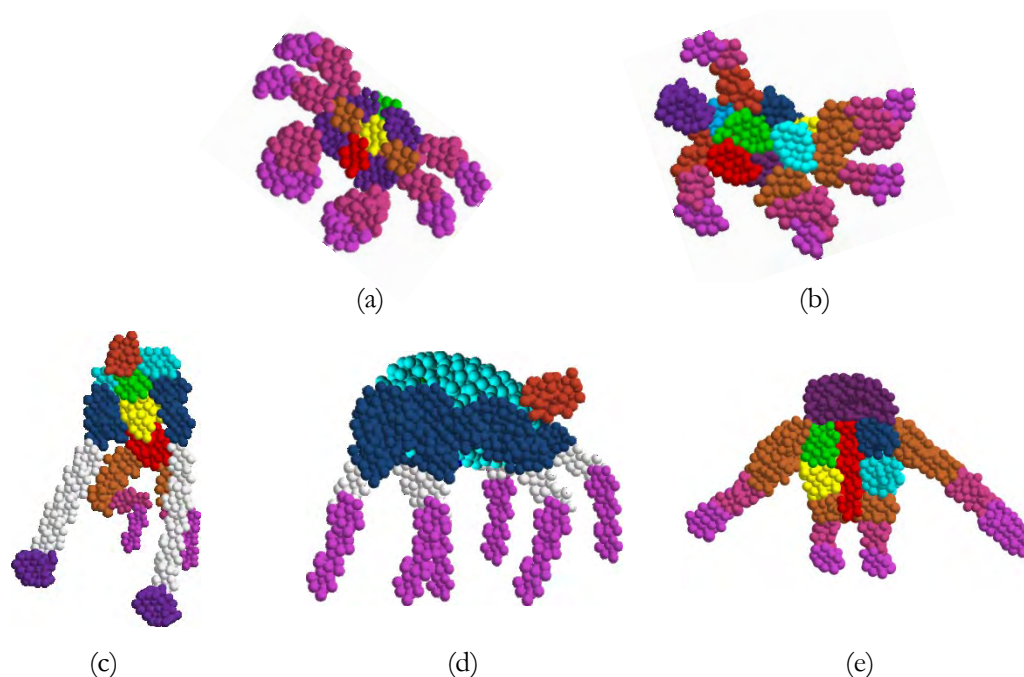
Se ha realizado un estudio consistente en una exploración paramétrica sencilla de las estructuras *MapDevo3D* en el sentido del polimorfismo mencionado anteriormente, donde ligeras variantes de un determinado genoma  $G$  producen ligeras variantes estructurales del fenotipo. En este caso, las variaciones son aún más modestas que las representadas en la Figura 5.11c, ya que son sólo cuantitativas y se refieren al tamaño del cuerpo y la longitud de las extremidades en un organismo de cuatro extremidades (el resto en el genotipo permanece igual). Lo que se evalúa, sin embargo, no es la morfología, sino el éxito de comportamiento ante el reto de subir escaleras. El valor de bondad mide la rectitud de la trayectoria seguida mediante el cálculo de la relación de la distancia en línea recta recorrida sobre la longitud del camino real. Los resultados que se muestran en la Figura 5.12 parecen indicar que los cuerpos más grandes funcionan mejor, mientras que puede haber un tamaño óptimo para las extremidades. Obviamente, más allá de estos ensayos de prueba de conceptos, se necesita una búsqueda evolutiva más amplia permitiendo modificaciones drásticas del plan corporal. Además el trabajo debe ser conducido hacia cómo un sistema embriomórfico puede evolucionar de forma espontánea, es decir, cómo puede sufrir variaciones aleatorias o no aleatoriamente seleccionadas en base a su éxito en la realización de ciertas tareas.



#### 5.4. Función a partir de la estructura, estructura a partir del desarrollo: Un proceso evolutivo doblemente indirecto

---

Por último, se han realizado otras pruebas experimentales sobre *MapDevo3D* en referencia a la especiación (Figura 5.11d) donde genomas drásticamente diferentes crean diferentes estructuras. La figura 5.13 muestra diferentes casos de especiación usando *MapDevo3D*. Estas morfologías han sido obtenidas mediante la creación de genotipos dirigidos de forma manual, sin proceso evolutivo, con la única intención de buscar morfologías interesantes, sin comportamiento funcional debido al alto coste computacional que supone su simulación física al estar algunos de los organismos compuestos por cerca de 6.000 células. Sin embargo, dan una muestra de las interesantes y variadas morfologías que se podrían obtener bien por simple búsqueda aleatoria o aplicando búsqueda evolutiva.



**Figura 5.13: Diferentes casos de especiación en *MapDevo3D*. Genotipos diseñados a mano con el objetivo de buscar fenotipos con morfologías interesantes. (a) Criatura con forma de cangrejo. (b) Reptil con 4 extremidades y cola (c) Criatura bípeda con lardos brazos (d) Especie de insecto con 6 extremidades (e) Criatura bípeda.**

Cabe destacar, que esta exploración del espacio de búsqueda de morfologías podría tener una útil aplicación en la búsqueda de formas de personajes o criaturas (existentes o no en la naturaleza) dentro del campo de la animación.

## 5.5. Conclusiones y discusiones

La ingeniería embriomórfica es intrínsecamente interdisciplinar, ya que sigue de cerca los principios biológicos a un nivel abstracto, pero no intenta modelar datos detallados de genomas u organismos reales. Por lo tanto, se asienta en la encrucijada entre diferentes dominios, desde el desarrollo y sistemas biológicos hasta la vida artificial, en particular, la computación amorfa/espacial [13], robótica de enjambres y programación evolutiva. Siguiendo los principios de la ingeniería morfogénica, constituye un intento inicial de dotar a un sistema físico con la información o, desde el punto de vista opuesto, integrar un sistema de información en un sistema físico [56]. Lo hace mediante la combinación de (1) auto-ensamblaje mecánico (*SA*) y (2) la formación de patrones computacionales (*PF*), bajo (3) el control de un programa genómico (*G*). En *MapDevo3D*, estos principios son modelados por procesos dinámicos, respectivamente: (1) la adhesión celular (por medio de fuerzas elásticas), (2) la difusión de morfógeno (a través de los contadores de gradiente), y (3) la expresión génica (a través de una red de regulación genética simple y esquemática).

Los modelos recientes de agentes controlados por genes basados en co-evolución (y co-desarrollo) de cuerpo y cerebro también han mostrado un camino prometedor hacia un enfoque *evo-devo* artificial totalmente integrado [95], [24], [156]. En última instancia, abstrayéndose más lejos del desarrollo biológico, un objetivo importante de la ingeniería embriomórfica es contribuir al diseño de nuevos sistemas de auto-organización capaces de sustituir a los arquitectos con grandes colectividades descentralizadas de agentes. Muchos trabajos de investigación han investigado la posibilidad de obtener propiedades de auto-formación a partir de una variedad de componentes de computación complejos: nano-unidades, bacterias, agentes de software, piezas de robot, mini-robots, y así sucesivamente [53]. Dado que la funcionalidad se distribuye en un gran número de componentes, sería una tarea inabordable ensamblar e instruir a cada uno de ellos individualmente. Más bien, de

una manera similar a las células biológicas, estos componentes deben ser fácilmente producidos en masa, inicialmente como copias idénticas, y adquirir sus posiciones y funciones especializadas por sí mismos dentro del sistema, una vez que estén mezclados entre sí. Las aplicaciones potenciales del modelo propuesto *MapDevo3D* pueden ser múltiples, abarcando desde la robótica colectiva (como es obvio por la propia naturaleza del modelo y como se ha mencionado durante el desarrollo del capítulo) principalmente orientada al campo de la *soft-robotics*, la biología sintética o la nanotecnología.

De forma paralela a su aplicación en robótica colectiva, las propias características técnicas, físico-mecánicas y funcionales del modelo presentado (criaturas compuestas por cientos de esferas sólidas unidas por resortes elásticos que son animadas en un mundo virtual regido por leyes físicas) permiten también pensar en una posible aplicación en el campo de la animación, encuadrándose dentro de la disciplina de la animación basada en física, concretamente en la animación de cuerpos blandos (*soft-bodies*), las cuáles se caracterizan por su gran aportación en cuanto a realismo gráfico de sus modelos. De una forma casi directa, *MapDevo3D* podría utilizarse como una herramienta de generación de personajes o criaturas animadas desarrolladas mediante programación genética y modular. Tanto la búsqueda aleatoria de genotipos  $G$  como la evolución de estos usando algoritmos genéticos permitirán obtener criaturas de variadas e inimaginables morfologías (Figura 5.13), las cuáles pueden servir de base a los animadores gráficos. Además de la propia morfología desarrollada *MapDevo3D* también ofrece, gracias a sus dinámicas físico-mecánicas, animaciones bastante realistas de estas morfologías contribuyendo de esta manera al campo de la creatividad computacional en lo referente a la producción automática de contenidos audiovisuales.



## Capítulo 6

### Descripción de contenidos y conclusiones

Este capítulo final presenta una descripción de los contenidos de cada una de las partes de este trabajo de tesis en la sección 6.1, mientras que las conclusiones son expuestas en la sección 6.2.

#### 6.1. Descripción de contenidos

El primer capítulo introduce la línea de investigación y las principales motivaciones de este trabajo de tesis. Se presentó la cuestión sobre la capacidad creativa de las computadoras y su estudio por parte del campo de la creatividad computacional. También fue introducido el nuevo campo de la biología evolutiva del desarrollo (*evo-devo*) así como sus implicaciones computacionales e influencias sobre nuevas estrategias adoptadas por los algoritmos evolutivos en el marco del desarrollo artificial. La innovación y diversidad requerida en los procesos creativos puede potenciarse mediante la inclusión del proceso de desarrollo en los algoritmos evolutivos, a través de la codificación indirecta del esquema genotipo-fenotipo. Adicionalmente, la ingeniería embriomórfica, un caso particular de la ingeniería morfogénica que comparte los principios de la biología evolutiva del desarrollo, fue presentada como disciplina que persigue el desarrollo de arquitecturas funcionales mediante la agregación descentralizada y la auto-organización de una gran cantidad de pequeños agentes en base a información codificada en un genoma.

En el capítulo 2 se presenta *Melomics*, un nuevo enfoque en creatividad computacional aplicado a la composición musical, el cual, en vez de las técnicas de

inteligencia artificial tradicionales, se basa en un paradigma *evo-devo* conectando dos dominios diferentes: la computación evolutiva con el desarrollo artificial y el lenguaje musical. Este enfoque proporciona formas muy poderosas para la codificación de piezas musicales y también abre el camino a una innovadora automatización de tareas creativas. Se han presentado *Iamus* y *Melomics109*, los dos sistemas informáticos que han aplicado la tecnología de *Melomics*, y se han descrito los hitos conseguidos por ambos supercomputadores dentro del campo de la composición musical. Desde el punto de vista de la creatividad computacional, una cuestión de gran importancia es si la música compuesta por la tecnología *Melomics* puede generar y transmitir las mismas emociones que la compuesta e interpretada por un humano. Con este objetivo se han presentado los resultados de un experimento controlado llevado a cabo que determinaron que la música generada por *Melomics* resulta indistinguible de la música compuesta por un humano, tanto desde el punto de vista emocional como evocador. Dentro de las aplicaciones de esta tecnología, el desarrollo de *M<sup>3</sup>*, un subsistema de *Melomics*, plantea el origen de una amplia e innovadora familia de herramientas terapéuticas. La eficacia de este sistema fue comprobada mediante la realización de un experimento en el ámbito pediátrico que determinó los beneficios *M<sup>3</sup>* en la reducción del dolor durante un procedimiento de punción cutánea. Este sistema puede ser utilizado para prevenir y tratar varias condiciones tales como la ansiedad, el estrés, dolor crónico y trastornos del sueño. El uso de esta tecnología podría mejorar la eficacia del tratamiento y reducir el uso de fármacos con la consecuente disminución de los efectos secundarios para los pacientes.

En el capítulo 3, se describe una herramienta de creatividad computacional para la animación de personajes guiados por música. Haciendo uso de la computación evolutiva y su capacidad para explorar el espacio de búsqueda de soluciones, se presenta una herramienta que puede servir como generador de ideas para animadores aficionados o profesionales. Esta herramienta puede suponer una gran

## 6.1. Descripción de contenidos

---

ayuda ahorrando tiempo y esfuerzo en el proceso de animación. Por un lado, al usar un enfoque que se sustenta en la disciplina de la animación de caracteres basada en física se obtienen soluciones aceptables y atractivas visualmente. Por otro lado, este enfoque computacional, al no estar una computadora sesgada ni condicionada por el conocimiento previo, permite explorar y obtener soluciones (en este caso formas de bailar) que un humano quizás no podría concebir o plantearse. El modelo propuesto aporta una solución aceptable situada entre el modelado de una animación utilizando captura de movimientos y las técnicas de *keyframing*.

El capítulo 4, propone un modelo de desarrollo artificial, inspirado en el formalismo de las gramáticas de grafos reguladas por cadenas, aplicado a la robótica evolutiva y de desarrollo para la búsqueda de formas y comportamientos creativos en agentes escaladores. Los resultados de las simulaciones confirman la adecuación del uso de este formalismo, que implementa una codificación indirecta, como sistema de desarrollo de estructuras. La integración de este modelo de desarrollo en un algoritmo evolutivo, potencia el descubrimiento de diseños y comportamientos innovadores en estructuras de carácter ingenieril, tales como sistemas robóticos que deben desempeñar la tarea de ascender por una pendiente. Aplicado al campo de la robótica, el modelo propuesto puede asistir al diseñador o ingeniero en el proceso de diseño de un robot, inspirando soluciones tanto a nivel hardware (forma) y software (sistema de control / comportamiento).

En el capítulo 5, se presenta un marco de aplicación de la ingeniería embriomórfica al desarrollo de sistemas auto-organizados artificiales. El modelo *MapDevo3D* propone principios para el desarrollo de agentes multicelulares auto-organizados capaces de crear morfologías precisas y funcionales de una manera puramente endógena. Siguiendo los principios de la ingeniería morfogenética, establece normas genéricas para la emergencia de estructuras programables no aleatorias que no son ni repetitivas ni impuestas por las condiciones externas. Estos principios son modelados en *MapDevo3D* por procesos dinámicos, respectivamente: (1) la adhesión

celular (por medio de fuerzas elásticas), (2) la difusión de morfógeno (a través de los contadores de gradiente), y (3) la expresión génica (a través de una red de regulación genética esquemática). Abstrayéndose más lejos del desarrollo biológico, el trabajo desarrollado contribuye al diseño de nuevos sistemas auto-organizados capaces de sustituir a los arquitectos e ingenieros mediante grandes colectividades descentralizadas de agentes. Las características físico-mecánicas de los agentes desarrollados permiten la aplicación potencial del modelo propuesto como herramienta de soporte creativo en los campos de la robótica y animación *soft-bodied*.

## 6.2. Conclusiones

La cuestión sobre la capacidad creativa de las computadoras permanece aún abierta y seguirá siendo objeto de debate durante las próximas décadas. Esto es en parte debido a que el dominio de la creatividad computacional es intrínsecamente multidisciplinar y engloba aspectos entrelazados de índole filosófica, artística y computacional. No obstante, el continuo avance tecnológico de las computadoras unido a la aplicación de enfoques de diversa naturaleza está permitiendo la consecución de interesantes resultados en este campo. Una gran mayoría de estos avances no persiguen el ambicioso, y puede que utópico, objetivo de dotar a las computadoras de comportamientos creativos totalmente independientes que imiten y sustituyan la creatividad humana, sino que se centran en diseñar herramientas hardware y software que asistan a los humanos en sus propios procesos creativos permitiéndoles ahorrar tiempo y esfuerzo.

Posiblemente no exista mayor ejemplo de herramienta creativa que la propia naturaleza. No hay nada más que observar la extraordinaria cantidad de diferentes formas corporales y comportamientos que presentan los seres vivos que habitan nuestro planeta. Es un hecho evidente que la evolución natural, actuando durante millones de años, ha contribuido al surgimiento de tal innovación y diversidad. Sin



embargo, hasta hace pocas décadas, se ha ignorado el papel fundamental de la evolución de los procesos de desarrollo regulados por el genoma, como componente clave para la evolución fenotípica de esta gran diversidad de comportamientos y diseños estructurales. Esta reflexión permite vislumbrar que la imitación de estos mecanismos presentes en la naturaleza, y su incorporación al campo de las ciencias de la computación, puede proporcionar potentes herramientas creativas que incorporen las dosis de innovación y diversidad propias de los seres vivos. En este sentido, el objetivo de esta tesis ha sido el estudio de la aplicación de un conjunto de disciplinas bioinspiradas tales como los algoritmos evolutivos, el desarrollo artificial y la ingeniería embriomórfica, inspiradas por la biología evolutiva del desarrollo, para modelar productos y procesos creativos en contextos tan diferentes como la composición musical, la medicina, la animación por computador y la robótica. Para ello se han propuesto y estudiado varios modelos computacionales enmarcados dentro de estas disciplinas y que sirven de soporte a la creatividad humana o que incluso pueden exhibir comportamiento creativo de forma independiente.

En los capítulos 2, 4 y 5 de esta tesis se han propuesto enfoques que estudian el papel del proceso de desarrollo en el surgimiento de diversidad e innovación propias de los productos creativos, permitiendo obtener fenotipos complejos a partir de genotipos relativamente sencillos. Esto es debido a que estos enfoques utilizan una codificación indirecta del esquema genotipo-fenotipo, de manera que el genotipo no mapea al fenotipo de forma directa sino que representa el proceso de desarrollo que da lugar a ese fenotipo. Por un lado, como se ha visto en el capítulo 2, estos enfoques han demostrado ser útiles para la evolución de estructuras musicales complejas indistinguibles de las creadas por un humano y con una capacidad similar para la generación de emociones en el oyente. Esta forma de codificar estructuras musicales permite automatizar el proceso de composición musical dando lugar a una amplia gama de herramientas tecnológicas con aplicación en diferentes campos,

entre ellos el de los sistemas de tratamiento terapéutico. Por otro lado, como refleja el capítulo 4, un enfoque de este tipo también se ha mostrado útil para el surgimiento de innovación y diversidad de morfologías y comportamientos de agentes robóticos que deben cumplir una determinada tarea. Además, como se ha demostrado en el capítulo 5, la imitación del proceso de desarrollo, codificado en un genoma, permitiría la generación de una gran diversidad de macro-estructuras descentralizadas formadas por cientos de agentes auto-organizados. La construcción de estructuras de este tipo con tal alto grado de granularidad, sin tener en cuenta el proceso de desarrollo, requeriría de un gran esfuerzo por parte del diseñador debido a la gran cantidad de componentes y relaciones existentes entre ellos que caracterizan a estructuras de este tipo. Por último, el capítulo 3, representa una muestra de que los algoritmos evolutivos, que hacen uso de una codificación directa, también son útiles para aportar soluciones a problemas de índole creativo en los que resulta complicado establecer una codificación indirecta entre genotipo y fenotipo.

Todos los modelos presentados en esta tesis tienen en común el hecho de servir de soporte a la creatividad del ser humano en diferentes dominios de aplicación, permitiéndole saltar varios pasos hacia delante y comenzar en un punto más avanzado del proceso creativo, a partir de la solución proporcionada por el modelo. De igual modo que el conocido “miedo a la hoja en blanco” al que se enfrentan los escritores, a veces uno de los mayores problemas que tienen que encarar los artistas o diseñadores es comenzar a componer o diseñar un producto creativo como puede ser un tema o pieza musical, una animación, o el hardware y software de un robot. Dependiendo del grado de exigencia de cada individuo las soluciones proporcionadas por los modelos propuestos en esta tesis pueden considerarse un producto final y ser utilizado directamente, aunque también pueden servir de meras ideas inspiradoras que den lugar a soluciones más complejas generados por el autor a partir de ellas. Como se mencionó en el capítulo de introducción, ambos aspectos son objetivos igualmente perseguidos en el campo de la creatividad computacional.

## 6.2. Conclusiones

---

Queda en el aire una reflexión acerca de la medida en la que los indicios de creatividad presentados por los modelos propuestos deben ser adjudicados a la propia computadora o al diseñador o ingeniero humano que ha programado los algoritmos de los cuales surge tal creatividad. Sin embargo, como se comenta al principio de esta sección, se trata de una cuestión que carece de una sencilla respuesta debido a las complejas implicaciones filosóficas y legales inherentes al campo de la creatividad computacional.



## Apéndice A

### Detalles de implementación hardware y software

Este apéndice recoge los diversos detalles del hardware y software utilizados en esta tesis. La sección A.1 presenta los principales aspectos del clúster de computación utilizado para realizar las simulaciones de los modelos computacionales presentados en los capítulos 4 y 5. La sección A.2 describe brevemente detalles sobre la implementación software de estos modelos de desarrollo evolutivo para su ejecución paralela en el clúster de computación. Finalmente, en la sección A.3, se presentan diversos aspectos sobre la metodología y software utilizados en el capítulo 3 para la importación en *Blender* de las coreografías de personajes obtenidas en la simulación física, y su posterior renderizado.

#### A.1. Configuración del clúster de computación

Debido a que gran parte de las simulaciones y los algoritmos evolutivos que se presentan en esta tesis tienen un coste computacional relativamente alto, se utilizó un clúster de computación que pertenece y es administrado por el personal del grupo de Estudios en Biomimética (GEB). Este clúster dispone de la siguiente configuración:

- Un nodo principal con 4 procesadores *AMD Opteron™ Processor 6128*, lo que hace un total de 32 núcleos, con una frecuencia 2 Ghz cada uno. Para poder dotar a cada núcleo de, al menos, 2 GB de memoria RAM, dispone de 64 GB de memoria DDR3 a 1333 Mhz, distribuida en 8 módulos de 8 GB de capacidad cada uno. Para el almacenaje de datos, se utilizan 5 discos duros de 2 TB de capacidad.

- 10 nodos esclavos con 4 procesadores *AMD Opteron™ Processor 6128*, dotados de 8 núcleos a 2 Ghz, que multiplicado por los diez nodos de cómputo, obtiene una potencia de cálculo de 640 Ghz, distribuida en 320 núcleos, y una capacidad RAM total de 640 GB. Cada nodo dispone de un disco duro de 6 TB GB como almacenamiento local para el sistema operativo y archivos temporales.
- Una red de datos para comunicar los nodos para cálculo y hacer accesible la unidad de almacenamiento del nodo maestro a todos los nodos esclavos. Esta red de datos se implementa como dos redes *Gigabit Ethernet*.
- Una red administrativa a efectos de la configuración y acceso de línea de comandos a cualquier nodo esclavo desde el nodo maestro. Esta red administrativa es implementada usando una red *Fast Ethernet*.

	Nodo Maestro	Nodos Cómputo	Total
<b>Procesadores AMD Opteron</b>	4	40	44
<b>Núcleos</b>	32	320	352
<b>Memoria RAM DDR3 (GB)</b>	64	640	704
<b>Almacenamiento bruto (TB)</b>	10	60	70
<b>Ancho de banda</b>	18	20	N/A

Tabla A.1: Visión global del clúster

El nodo principal también tiene otro enlace de red conectado a la intranet de la universidad, con el fin de ser utilizado de forma remota. En total, el clúster dispone de 352 CPUs y 704 GB de la memoria principal. Todos los nodos del clúster ejecutan un sistema operativo *Debian GNU/Linux*. La planificación de tareas es implementada mediante el administrador de recursos *TORQUE* y el planificador *MAUI*, con el fin de balancear la carga de computación en el clúster. *TORQUE* fue

configurado para gestionar un grupo de 348 unidades de cómputo (320 CPUs de los nodos esclavos, y 28 CPUs más del nodo maestro). Las restantes 4 CPU en el nodo maestro se dejaron para controlar procesos en el nodo maestro y diversas tareas administrativas.

En resumen, el clúster se ha configurado para ser utilizado de la siguiente manera: conectarse de forma remota al nodo principal a través de *SSH*, y ejecutar el control de procesos en el mismo, que a su vez genera procesos computacionales concurrentes en los nodos esclavos, usando *TORQUE / MAUI*. Además, los procesos de computación se podrían ejecutar directamente en los nodos esclavos, accediendo a ellos desde el nodo maestro por *SSH*. La aplicación *screen GNU* estaba disponible en todos los nodos como medio para mantener las sesiones ejecutándose mientras el usuario no estaba conectado, y *Ganglia5* se utilizó para controlar la carga computacional de los nodos del clúster (en términos de uso de la memoria, ancho de banda y CPU).

## A.2. Implementación de los modelos computacionales

Para la implementación de los modelos presentados en los capítulos 3, 4 y 5, se utilizó el lenguaje de programación *Python*, debido a la rapidez que ofrece en la implementación de prototipos, así como su amplia variedad de librerías de simulación física y visualización gráfica. En nuestro caso, se ha utilizado el motor de física *ODE* proveído por la librería *PyODE* <sup>3</sup> para la simulación física; y la librería *VPython* <sup>4</sup> para la implementación gráfica.

El modelo evolutivo presentado en el capítulo 3, debido a su carácter interactivo y al no requerir una especial capacidad de cómputo, fue ejecutado en una sola

---

<sup>3</sup> <http://pyode.sourceforge.net/>

<sup>4</sup> <http://vpython.org/>

computadora de forma local. Esta computadora disponía de procesador *Intel Core 2 Quad* a 2.66 GHz y 8 GB de memoria RAM.

Por su parte, la implementación de los algoritmos evolutivos y simulaciones físicas utilizados en los capítulos 4 y 5 (sobre todo el primero de ellos que tiene una carga evolutiva mayor), al requerir una alta capacidad de cómputo y paralelización al incluir un proceso de desarrollo, utilizan el diseño maestro-esclavo del clúster de computación descrito en la sección anterior. El código que implementa el ciclo del algoritmo evolutivo, con excepción de la evaluación de los individuos (agentes u organismos), fue ejecutado directamente en el nodo maestro, debido al bajo coste computacional de las tareas de inicialización, selección, reproducción, cruce y mutación. La evaluación de los individuos de una generación (cálculo del valor de bondad) era distribuida como trabajos independientes que eran ejecutados en paralelo en los nodos esclavos. De esta forma, el ciclo para obtener una nueva generación en el algoritmo evolutivo consistía en (1) reproducción, cruce (en el caso de que se implemente) y mutación de la población actual en el nodo maestro y; (2) ejecución en paralelo de las simulaciones de cada individuo para el cálculo de su valor de bondad. El cálculo del valor de bondad de cada individuo se realiza en los nodos esclavos en dos fases. Primero, se calcula el fenotipo del individuo mediante el proceso de desarrollo determinado por el genotipo. Una vez que el individuo se ha desarrollado, éste es sometido a la simulación física obteniendo un valor de bondad en función de cómo haya ejecutado la tarea correspondiente, siendo este valor devuelto al nodo maestro. En condiciones normales, este proceso generaría un alto tráfico en la red del clúster de computación debido al continuo envío de datos entre nodo maestro y esclavos. Sin embargo, en nuestro caso se evita este problema debido a que sólo se envía a los nodos esclavos el genoma (una cadena de pocos bytes), siendo dentro de los nodos esclavos donde se desarrolla el individuo (cuyo peso computacional es bastante mayor que el del genoma).



### A.3. Renderización de animaciones de personajes en *Blender*

En esta sección se recogen algunos aspectos de la herramienta propuesta en el capítulo 3 para la sincronización automática de personajes y música en escenas de animación. Como se comenta en el capítulo 3, se ha utilizado el software de animación *Blender* para importar los datos tanto de estructura del cuerpo del personaje (*cuerpo.txt*) como de su animación (*animacion.txt*) obtenidos en la simulación física llevada a cabo usando el motor de física *ODE*. El contenido típico del archivo *cuerpo.txt* es el que se muestra en la Figura A.1, en el que las tres primeras columnas indican los valores de las coordenadas x, y, z, para cada uno de

```

1 0.000167220836318 0.20001502224 1.10001929036 PIE_DER_BASE
2 0.400486314158 0.0999891936388 1.00009732501 PIE_DER_DEDO1
3 0.400214995639 0.0999813573689 1.20001170326 PIE_DER_DEDO2
4 -0.100078600029 0.0999932738803 1.00004818162 PIE_DER_DEDO3
5 -0.0999885805028 0.0999965632815 1.20004796486 PIE_DER_DEDO4
6 0.000136771430585 0.199985609298 1.89995083786 PIE_IZQ_BASE
7 0.399922262097 0.0999968276563 1.79995701239 PIE_IZQ_DEDO1
8 0.399990506463 0.0999989971647 1.9998007409 PIE_IZQ_DEDO2
9 -0.099947586839 0.0999960249434 1.7999390305 PIE_IZQ_DEDO3
10 -0.0999251097248 0.0999872025485 1.99993172511 PIE_IZQ_DEDO4
11 0.10124897597 1.20002641169 1.09996162123 RODILLA_DER
12 0.0999512379926 1.19999169845 1.90002084944 RODILLA_IZQ
13 -0.00203614653458 2.20014832866 1.19948083726 CADERA_DER
14 0.000883437162246 2.19991890916 1.80016419588 CADERA_IZQ
15 0.00941982111245 2.19995150723 1.49987902213 CADERA_CEN
16 0.000534560022245 3.39995480114 0.999696536499 HOMBRO_DER
17 -0.000754792273669 3.40011512424 2.00032654293 HOMBRO_IZQ
18 8.13966751444e-05 3.49943316507 1.49997176403 PECHO
19 -0.0165057517671 2.83679490066 0.578818219698 CODO_DER
20 -0.0150518851399 2.83650958782 2.42122750781 CODO_IZQ
21 0.0687672170877 2.34952034625 0.219801737338 MANO_DER
22 0.0681526329009 2.24866417323 0.0191081523308 MANO_DER_DEDO1
23 0.0598250546482 2.12872517352 0.179967153772 MANO_DER_DEDO2
24 0.069603041147 2.34939123956 2.78031374429 MANO_IZQ
25 0.068794074472 2.24848507304 2.98116913443 MANO_IZQ_DEDO1
26 0.0606584513228 2.12860960099 2.82029271086 MANO_IZQ_DEDO2
27 0.000161864314225 4.10007283425 1.49992523081 CABEZA

```

Figura A.1: Contenido del fichero *cuerpo.txt*. Las tres primeras columnas representan los valores de las coordenadas x, y, z para cada uno de los puntos de unión del cuerpo determinados en la cuarta columna.

los puntos de unión del cuerpo definidos en la cuarta columna. El contenido del fichero *animacion.txt* es muy similar al del fichero *cuerpo.txt*, con el añadido de que contiene un número considerablemente mayor de filas, las cuales representan la posición en cada instante de la animación de los puntos de unión del cuerpo del personaje. A modo de ejemplo, siguiendo con la estructura del fichero *cuerpo.txt*, las primeras 27 líneas del fichero *animacion.txt* representarían la posición de los puntos de unión del cuerpo en el instante  $t=0$ , otras 27 líneas para representar su posición en  $t=2$ , y así sucesivamente hasta contener la posición de todos los instantes de la animación.

La importación en *Blender* de los datos del cuerpo y la animación obtenidos en la simulación física se realiza de forma automática mediante dos *scripts* escritos en *Python* (*importArmadura.py* e *importAnimacion.py*), programados para tal objetivo. La Figura A.2 muestra un extracto del script *importArmadura.py* en el editor de *scripts* *Python* que *Blender* proporciona. Este script utiliza la información contenida en el

```

109 f = open("../cuerpo.txt")
110
111 indicey=2
112 indicey=0
113 indicez=1
114
115 infile=f.readlines()
116
117 var=infile[0].split(" ")
118 x=float(var[indicey])
119 y=float(var[indicey])
120 z=float(var[indicez])
121 PIE_DER_BASE = (x,y,z)
122
123 var=infile[1].split(" ")
124 x=float(var[indicey])
125 y=float(var[indicey])
126 z=float(var[indicez])
127 PIE_DER_DEDO1 = (x,y,z)
128
129 var=infile[2].split(" ")
130 x=float(var[indicey])
131 y=float(var[indicey])
132 z=float(var[indicez])
133 PIE_DER_DEDO2 = (x,y,z)
    
```

Figura A.2: Extracto del script *importArmadura.py*, el cual genera una armadura en *Blender* a partir del contenido del fichero *cuerpo.txt*.

### A.3. Renderización de animaciones de personajes en *Blender*

archivo *cuerpo.txt*, creando automáticamente la armadura del personaje en *Blender* como se aprecia en la Figura A.3. Una vez hecho esto, el siguiente paso consiste en dotar a la armadura de la animación obtenida en la simulación física. Para ello, se selecciona la armadura que se acaba de crear y se ejecuta el *script importAnimacion.py* de igual manera que el anterior, el cual incorpora los datos



Figura A.3: Importación de la armadura en *Blender* usando el *script importArmatura.py*.

de la animación a la armadura como se puede apreciar en la Figura A.4. En la parte inferior de esta captura se puede observar en color amarillo como se han creado automáticamente los *keyframes* de la animación. El siguiente paso consiste en añadir a la animación el tema musical que se utilizó para guiar la animación física, utilizando la opción editor de secuencias de video disponible en *Blender*, como se aprecia en la parte inferior de la Figura A.5. Una vez hecho esto, al reproducir la animación se puede comprobar cómo la animación de la armadura está sincronizada con el tema musical. La siguiente tarea a realizar es el denominado proceso de *binding*, mediante

el cual dotamos a nuestro personaje de una textura. Existen multitud de texturas libres disponibles en internet. Concretamente la textura del personaje de *Spiderman* utilizada en las ilustraciones del capítulo 3 han sido obtenidas del portal *Blendswap*<sup>5</sup>. La Figura A.6, muestra detalles del proceso de *binding* entre la textura seleccionada y la armadura animada. Si se ha realizado el proceso de *binding* de forma adecuada, al reproducir la animación se puede observar como la textura acompaña los movimientos de la armadura animada, tal y como se aprecia en la Figura A.7. Por último, para dar más vistosidad a la animación, se puede introducir el personaje en un escenario de los muchos gratuitos que se pueden obtener en el portal *Blendswap*, tal y como se muestra en la Figura A.8.

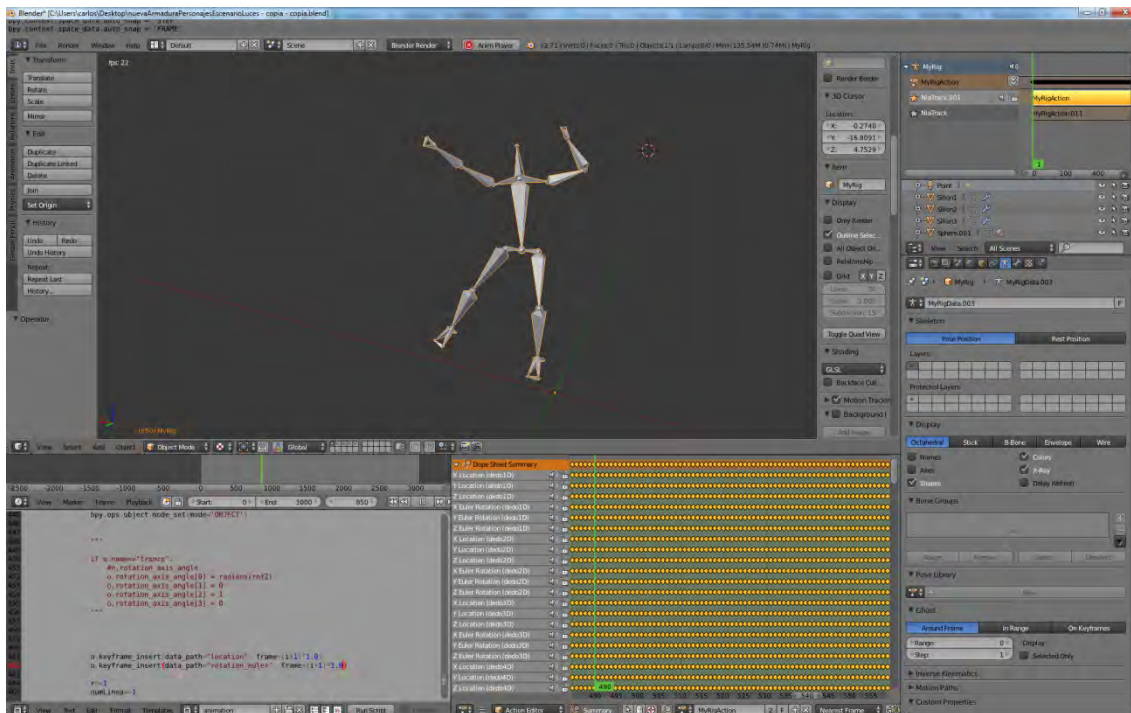


Figura A.4: Importación automática de la animación en la armadura previamente creada. En color amarillo se pueden apreciar los keyframes generados para la animación.

<sup>5</sup> <http://www.blendswap.com>

### A.3. Renderización de animaciones de personajes en *Blender*

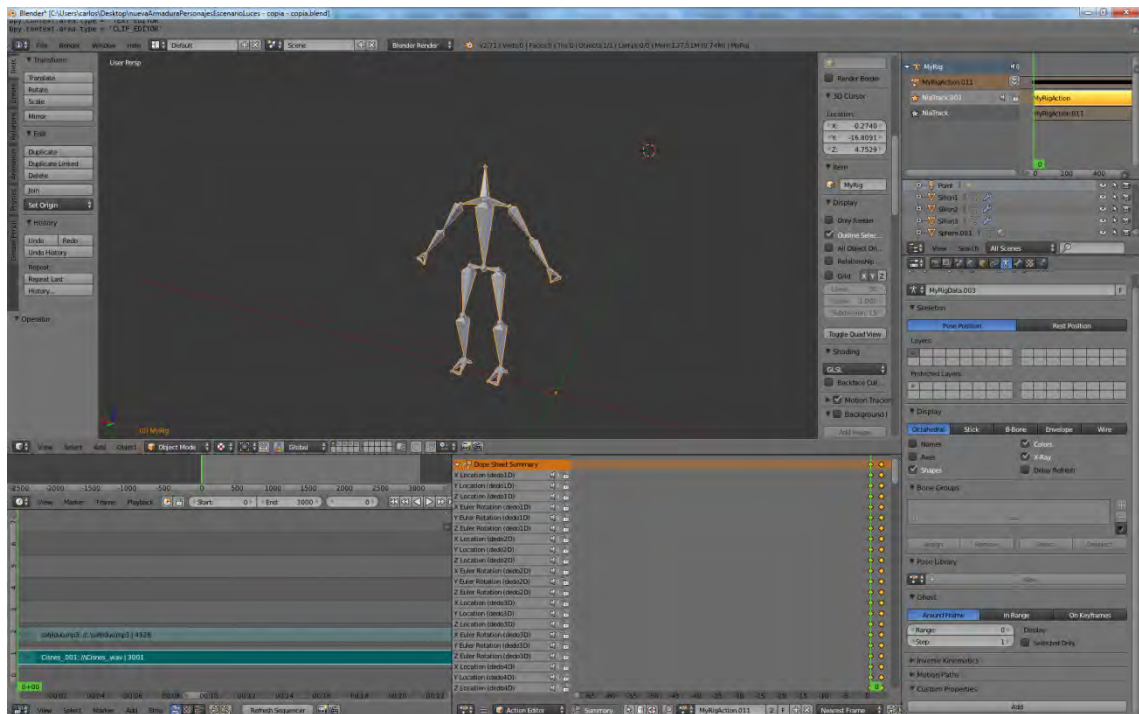


Figura A.5: Importación de tema musical en la animación mediante el Editor de Secuencias de Video disponible en *Blender*.

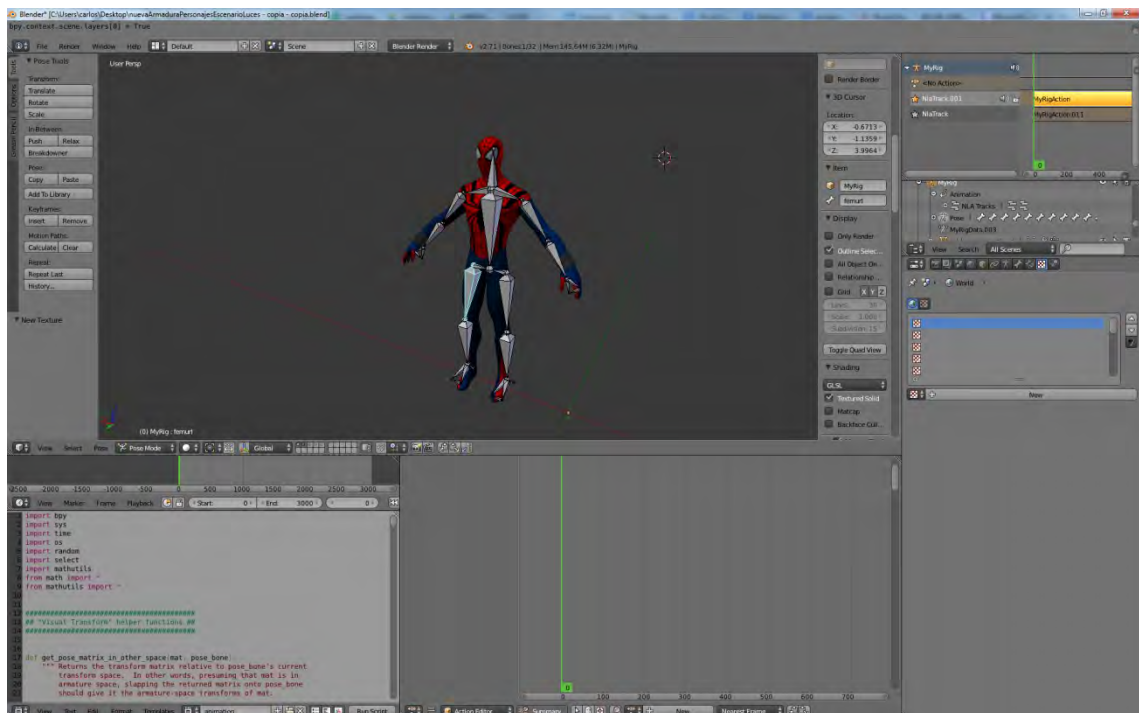


Figura A.6: Proceso de *binding* mediante el cual se incorpora una textura a la armadura

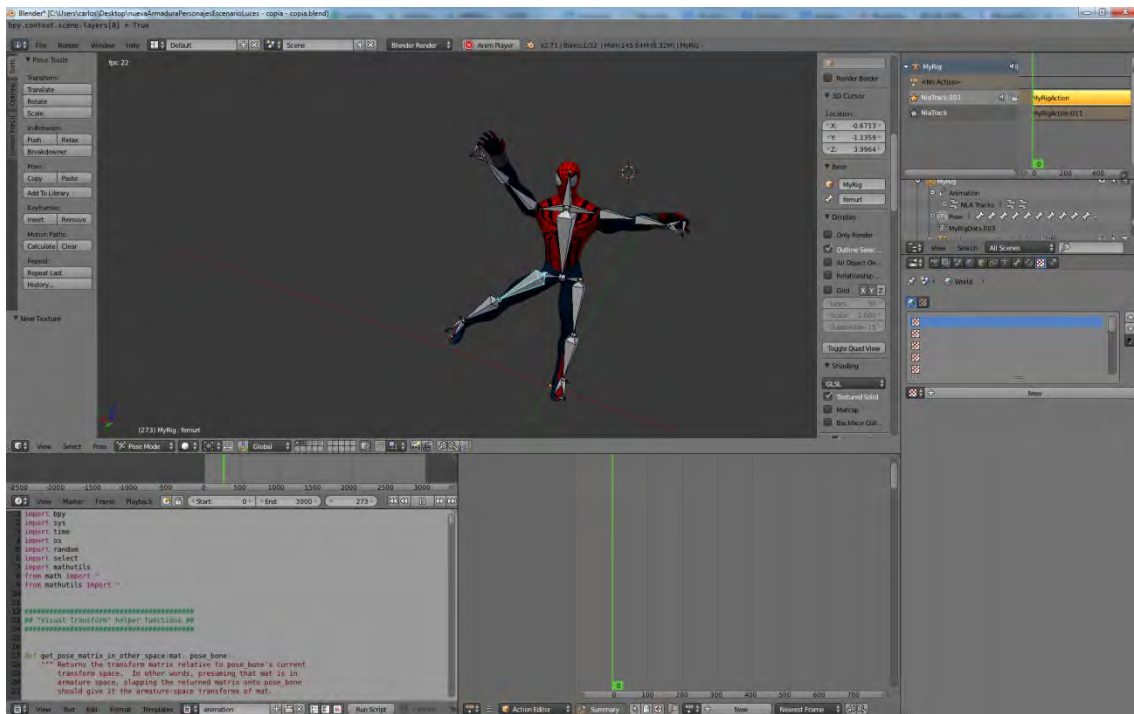


Figura A.7: Animación de armadura y textura tras el proceso de *binding*. Se puede apreciar como la textura acompaña los movimientos de la animación de la armadura.

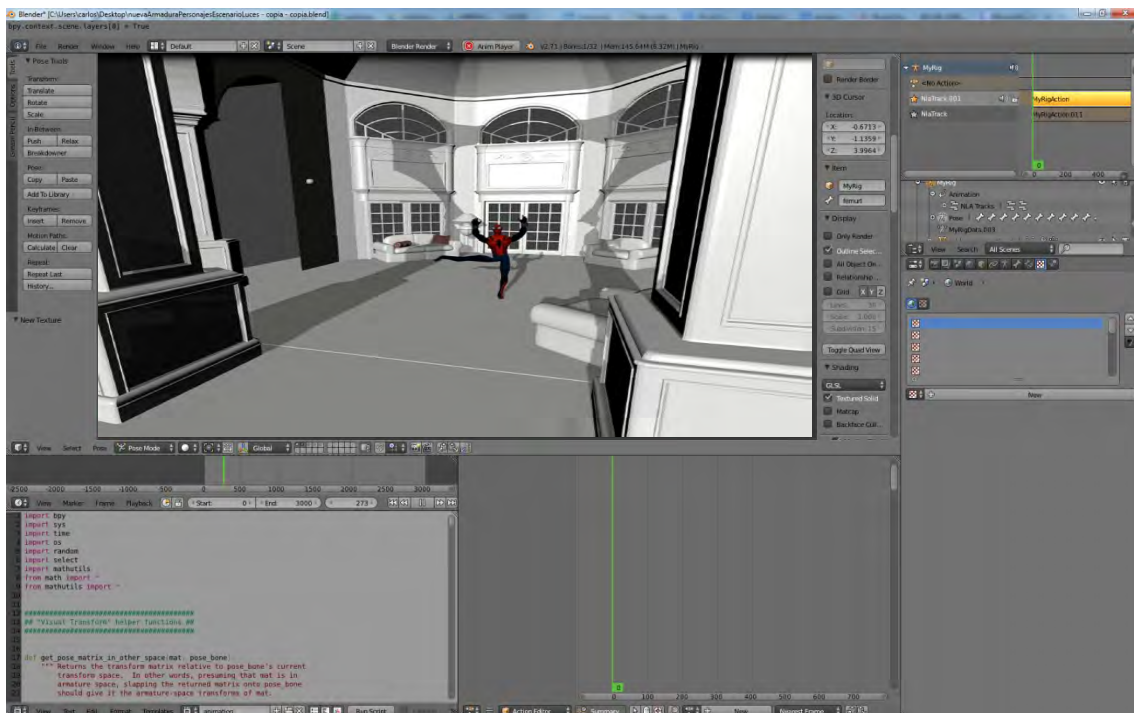


Figura A.8: Inclusión del personaje animado en una escena.

## Apéndice B

### Formularios y documentos utilizados en los experimentos realizados

Este apéndice recoge los formularios utilizados en los dos experimentos realizados relacionados con la tecnología *Melomics*. Por una parte, la sección B.1 recoge detalles sobre la interfaz del cuestionario interactivo usado en el experimento realizado para validar el uso potencial de *Melomics* como sistema de generación musical, estudiando la comparación entre la música generada por el supercomputador *Melomics109* y la creada por un compositor humano. La sección B.2, incluye la documentación y cuestionarios utilizados en el experimento llevado a cabo para validar el uso potencial de *Melomics* como sistema de tratamiento terapéutico.

#### **B.1. Experimento para estudiar la comparación entre la música generada por *Melomics* y la creada por seres humanos**

A continuación se muestran las diferentes capturas de pantalla de la aplicación interactiva utilizada para recoger los datos de los sujetos participantes en el experimento.



## Consentimiento informado para participación en experimento

Le pedimos su colaboración para participar en la presente investigación llevada a cabo por la Universidad de Málaga en colaboración con el Museo Interactivo de la Música de Málaga (MIMMA). La meta de este experimento es estudiar cómo sentimos el sonido, qué efectos produce en nosotros y cómo activa nuestro cerebro.

Si usted accede a participar en este estudio escuchará 5 registros sonoros y se le pedirá completar un breve cuestionario para cada uno. Esto tomará aproximadamente 10 minutos de su tiempo.

La participación en este estudio es **TOTALMENTE VOLUNTARIA Y ANÓNIMA**. Toda la información recogida será tratada según la Ley de Protección de Datos de Carácter Personal y no se usará para ningún otro propósito fuera de los de esta investigación.

Si tiene alguna duda sobre este experimento, puede hacer preguntas en cualquier momento durante su participación en él. Igualmente, puede retirarse del experimento en cualquier momento.

Por favor, si decide aceptar y participar en este experimento, pulsa en el botón "Continue" situado al final de la página.

LE AGRADECEMOS SU PARTICIPACIÓN



## Informed consent to participate in experiment

You are invited to join a research study carried out by University of Málaga in collaboration with the Málaga Interactive Museum of Music (MIMMA). The main goal of this experiment is to study how we feel the sound, what effects it does have on us, and in what way does it activate our brain.

By accepting to participate in this study, you will be prompted to listen to 5 sound recordings, and you will be required to fill in a short questionnaire for each one. This should take about only 10 minutes of your time.

The participation in this study is **COMPLETELY VOLUNTARY AND ANONYMOUS**. All the collected information will be treated confidentially and we are committed to ensuring compliance with data protection or privacy legislation.

If you have any doubt or question about this study, you can ask them now or any time during the experiment. Of course, you can leave the study at any time.

Please take the time you need to read and understand the information in this form before make your decision. If you decide to join, please click "Continue" at the end of this page.

THANKS YOU FOR YOUR PARTICIPATION

« Back

Continue »



B.1. Experimento para estudiar la comparación entre la música generada por *Melomics* y la creada por seres humanos

---

\* Required

## Información de participante

Participant's information

**Edad \***

Age

**Sexo \***

Sex

Hombre / Male

Mujer / Female

**País \***

Country

**Correo electrónico (Opcional)**

Email Address (Optional)

**¿Posee algún tipo de formación académica musical igual o superior a 5 años? \***

Do you have at least 5 years of academic musical training?

Sí / Yes

No / No

« Back

Continue »



37% completed

\* Required

## Registro sonoro 1 de 5 / Sound recording 1 of 5

Pulse "play" para escucharlo / Please click "play" to listen it



Por favor, tras escuchar este registro sonoro, rellene las cuestiones que se le plantean de la forma más sincera posible, gracias. / Please, after listening, fill in the following questions as honestly as possible. Thank you!

**¿Diría que lo que ha escuchado es música? \***

Would you say that what you have listened to is music?

- Sí / Yes  
 No / No

**¿Genera representaciones mentales al escucharlo? \***

Does it generate mental representations when you listen to it?

- Sí / Yes  
 No / No

**En caso afirmativo, ¿podría escribir tres representaciones mentales que hayan venido a su mente mientras lo escuchaba?**

If so, could you write three mental representations that have come to mind listening to it?

**¿Genera estados emocionales? \***

Does it generate any feelings?

- Sí / Yes  
 No / No

**En caso afirmativo, ¿podría escribir tres sentimientos que haya sentido mientras lo escuchaba?**

If so, could you write three feelings that you have felt while listening to it?

« Back

Continue »

50% completed

## B.1. Experimento para estudiar la comparación entre la música generada por *Melomics* y la creada por seres humanos

\* Required

### Registro sonoro 2 de 5 / Sound recording 2 of 5

Pulse "play" para escucharlo / Please click "play" to listen it



Por favor, tras escuchar este registro sonoro, rellene las cuestiones que se le plantean de la forma más sincera posible, gracias. / Please, after listening, fill in the following questions as honestly as possible. Thank you!

**¿Diría que lo que ha escuchado es música? \***

Would you say that what you have listened to is music?

- Sí / Yes  
 No / No

**¿Genera representaciones mentales al escucharlo? \***

Does it generate mental representations when you listen to it?

- Sí / Yes  
 No / No

**En caso afirmativo, ¿podría escribir tres representaciones mentales que hayan venido a su mente mientras lo escuchaba?**

If so, could you write three mental representations that have come to mind listening to it?

**¿Genera estados emocionales? \***

Does it generate any feelings?

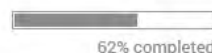
- Sí / Yes  
 No / No

**En caso afirmativo, ¿podría escribir tres sentimientos que haya sentido mientras lo escuchaba?**

If so, could you write three feelings that you have felt while listening to it?

« Back

Continue »



62% completed

\* Required

## Registro sonoro 3 de 5 / Sound recording 3 of 5

Pulse "play" para escucharlo / Please click "play" to listen it



Por favor, tras escuchar este registro sonoro, rellene las cuestiones que se le plantean de la forma más sincera posible, gracias. / Please, after listening, fill in the following questions as honestly as possible. Thank you!

**¿Diría que lo que ha escuchado es música? \***

Would you say that what you have listened to is music?

- Sí / Yes  
 No / No

**¿Genera representaciones mentales al escucharlo? \***

Does it generate mental representations when you listen to it?

- Sí / Yes  
 No / No

**En caso afirmativo, ¿podría escribir tres representaciones mentales que hayan venido a su mente mientras lo escuchaba?**

If so, could you write three mental representations that have come to mind listening to it?

**¿Genera estados emocionales? \***

Does it generate any feelings?

- Sí / Yes  
 No / No

**En caso afirmativo, ¿podría escribir tres sentimientos que haya sentido mientras lo escuchaba?**

If so, could you write three feelings that you have felt while listening to it?

« Back

Continue »



75% completed

B.1. Experimento para estudiar la comparación entre la música generada por *Melomics* y la creada por seres humanos

---

\* Required

## Registro sonoro 4 de 5 / Sound recording 4 of 5

Pulse "play" para escucharlo / Please click "play" to listen it



Por favor, tras escuchar este registro sonoro, rellene las cuestiones que se le plantean de la forma más sincera posible, gracias. / Please, after listening, fill in the following questions as honestly as possible.  
Thank you!

**¿Cree que esta pieza ha sido compuesta por un humano o por una computadora? \***

Do you think that the piece you have listened to has been composed by a composer or by a computer?

- Humano / Human
- Computadora / Computer

« Back

Continue »



87% completed

\* Required

## Registro sonoro 5 de 5 / Sound recording 5 of 5

Pulse "play" para escucharlo / Please click "play" to listen it



Por favor, tras escuchar este registro sonoro, rellene las cuestiones que se le plantean de la forma más sincera posible, gracias. / Please, after listening, fill in the following questions as honestly as possible.  
Thank you!

**¿Cree que esta pieza ha sido compuesta por un humano o por una computadora? \***

Do you think that the piece you have listened to has been composed by a composer or by a computer?

- Humano / Human
- Computadora / Computer

**Pulse Enviar/Submit para finalizar el experimento. Le agradecemos su participación.**

Click Enviar/Submit to finish the experiment. Thanks you for your participation.

« Back

Submit



100%: You made it

*Never submit passwords through Google Forms.*

B.2. Experimento para validar el uso potencial de *Melomics* como sistema de tratamiento terapéutico

---

## **B.2. Experimento para validar el uso potencial de *Melomics* como sistema de tratamiento terapéutico**

Esta sección recoge dos de los documentos utilizados en la realización del estudio:

- Información para el paciente y consentimiento informado.
- Cuaderno de recogida de datos

### **B.2.1. INFORMACION PARA EL PACIENTE Y CONSENTIMIENTO INFORMADO**

Título del estudio: **UTILIDAD DE MUSICOTERAPIA MEDIANTE USO DE APLICACIÓN MÓVIL MELOMICS® EN PRUEBAS DE PRICK TEST EN ALERGOLOGIA**

Código del estudio: Melomics @life

Centro del estudio: Hospital Materno Infantil Carlos Haya, Málaga.

**Antes de proceder a la firma de este consentimiento informado, léalo atentamente y realice las preguntas que considere oportunas.**

#### **Naturaleza:**

La Consulta de Alergología del Hospital Materno Infantil, en colaboración con el grupo de Biomimética, Proyecto *Melomics*, vinculado a la Universidad de Málaga, trabaja para mejorar la calidad de vida de los pacientes sometidos a pruebas potencialmente molestas y/o dolorosas, razón por la cual, pondrá en funcionamiento un programa piloto de estimulación, durante la realización de las pruebas de alergia para niños de entre 5 y 14 años, denominado “*Test cutáneos y estimulación*”.

**Importancia:**

La búsqueda de alternativas terapéuticas que promuevan el bienestar de los pacientes infantiles, y alivien su sufrimiento en pruebas potencialmente molestas y/o dolorosas, constituye un objetivo prioritario dentro del mundo de la medicina pediátrica. Iniciativas como la de este Centro Hospitalario ya han sido llevadas a cabo con éxito en otros países, sentando bases esperanzadoras.

**Riesgos para usted o para su hijo:**

Ninguno.

Yo, \_\_\_\_\_ (*nombre* \_\_\_\_\_ y \_\_\_\_\_ *apellidos*)  
..... en  
nombre de mi hijo (*nombre* \_\_\_\_\_ y \_\_\_\_\_ *apellidos* del *menor*)  
.....

He leído el documento informativo que acompaña a este consentimiento. Comprendo que la colaboración de mi hijo/a es voluntaria y soy libre para que participe o no en el programa piloto y que ***puedo retirarle cuando quiera y sin dar explicaciones.*** Se me informa que todos los datos obtenidos serán confidenciales y se tratarán conforme establece la Ley Orgánica de Protección de Datos de Carácter Personal 15/99. Se me informa que los resultados obtenidos sólo se utilizarán para fines clínicos y terapéuticos.

Por tanto, presto libremente mi conformidad para que mi hijo/a participe en el *proyecto titulado “Test cutáneos y estimulación”.*

Firma del participante (Representante legal del menor).....




Firma del profesional (Informador)..... Fecha: .....



## B.2. Experimento para validar el uso potencial de *Melomics* como sistema de tratamiento terapéutico

---

### Procedimiento.

1. Antes de iniciar la prueba, debe tocar la pantalla del teléfono para activarlo.
2. Pulse en el icono de @life.  Se abrirá la aplicación. Pulse en el dibujo del osito  y luego sobre el dibujo del robot.  La música se pondrá en marcha.
3. Escuche por los cascos. Ponga el volumen más adecuado para la edad del niño.
4. Colóquele los cascos al niño. Debe mantener la escucha de la música durante toda la prueba.
- 5, Una vez finalizada la prueba, pare la música y devuelva los cascos y los auriculares al personal.

**NOTA:** El teléfono cuenta con un dispositivo de geolocalización para garantizar su uso continuado durante todo el tiempo que dure el proyecto. Por favor, no abandone la sala de espera durante la prueba.

---

### PARA EL PERSONAL DEL HOSPITAL.

Número de Identificación *(a rellenar por el personal del hospital)*

Edad *(a rellenar por el personal del hospital)*

## ESCALA DE VALORACION VISUAL DEL DOLOR.-

*Para niños de entre 5 y 9 años.*



*Para niños de entre 10 y 14 años*



### B.2.2. CUADERNO DE RECOGIDA DE DATOS

#### DATOS DEMOGRAFICOS

1.- Edad:

2.- Sexo: Varón- Mujer

#### DATOS CLINICOS

1- VALOR DE ESCALA ANALÓGICA VISUAL adaptada según edad.

## Apéndice C

### Publicaciones derivadas de la tesis doctoral

- Requena, G., Sánchez, C., Corzo-Higueras, J. L., Reyes-Alvarado, S., Rivas-Ruiz, F., Vico, F. and Raglio, A. (2014), Melomics music medicine (M<sup>3</sup>) to lessen pain perception during pediatric prick test procedure. *Pediatric Allergy and Immunology*. DOI: 10.1111/pai.12263 (FACTOR DE IMPACTO (JCR 2013) – QUARTIL (JCR): 3.859 - Q1).
- Doursat, R., Sánchez, C. (2014) Growing fine-grained multicellular robots. *Soft Robotics*. 1(2): 110-121. DOI: 10.1089/soro.2014.0014.
- Sánchez C, Moreno F, Albarracín D, Fernández JD, Vico FJ (2013) Melomics: A Case-Study of AI in Spain. *AI Magazine*, 34(3), 99-103. DOI: <http://dx.doi.org/10.1609/aimag.v34i3.2464>. (FACTOR DE IMPACTO (JCR 2012) – QUARTIL (JCR): 0.726 – Q3).
- Doursat, R., Sánchez, C., Dordea, R., Fourquet, D. & Kowaliw, T. (2013) Chapter 11: Embryomorphic Engineering: Emergent innovation through evolutionary development. En *Morphogenetic Engineering: Toward Programmable Complex Systems*, R. Doursat, H. Sayama & O. Michel, eds. "Understanding Complex Systems" Series, Springer-Verlag, ISBN 978-3-642-33901-1. DOI: 10.1007/978-3-642-33902-8\_11.
- Francisco J. Vico, Carlos A. Sánchez-Quintana, David D. Albarracín (2011). MELOMICS: Aportaciones de la informática y la biología a la musicoterapia receptiva. En *Sabbatella, P. (coord.), Identidad y Desarrollo Profesional del Musicoterapeuta en España*, pag. 83-93. *Actas del III Congreso Nacional de Musicoterapia*. Cádiz 8-10 Octubre 2010. Granada: Grupo Editorial Universitario. ISBN: 978-84-9915-629-3



## Bibliografía

- [1] Abrams B., "Evidence-based music therapy practice: an integral understanding," *J Music Ther*, vol. 47, pp. 351-79, 2010.
- [2] Alankus G., Bayazit A.A., Bayazit B., "Automated motion synthesis for dancing characters," *Comput. Animation*, vol. 16(3-4), pp. 259-271, 2005.
- [3] Altshuler E., Linden D., "Design of a Wire Antenna Using a Genetic Algorithm," *Journal of Electronic Defense*, pp. 50-52, 1997.
- [4] Ames C., "Automated composition in retrospect: 1956-1986," *Leonardo*, vol. 20(2), pp. 169-185, 1987.
- [5] Armstrong W., Green M., "The dynamics of articulated rigid bodies for purposes of animation," *The visual computer*, pp. 231-240, 1985.
- [6] Auberbach J.E., Bongard J.C., "On the relationship between environmental and morphological complexity in evolved robots," *En Actas del 2012 Genetic and Evolutionary Computation Conference*, 2012, pp. 521-528.
- [7] Autumn K., "Gecko Adhesion: Structure, Function, and Applications," *MRS Bulletin*, vol. 32, pp. 473-478, 2007.
- [8] Back T., Hoffmeister F., Schewfwl H.P., "A survey of evolution strategies," *En Actas del 4th International Conference on Genetic Algorithms*, 1991, pp. 2-9.
- [9] Ball P., "Computer science: Algorithmic rapture," *Nature*, vol. 488(7412), p. 458, 2012.
- [10] Ball P., "The Self-Made Tapestry," *Oxford University Press*, 1999.
- [11] Barbic J., Da Silva M., Popovic J., "Deformable object animation using reduced optimal control," *ACM Trans. on Graphics (SIGGRAPH 2009)*, vol. 28, p. 3, 2009.
- [12] Barbic J., Popovic J., "Real-time control of physically based simulations using gentle forces," *ACM Trans. on Graphics (SIGGRAPH Asia 2008)*, vol. 27(5), pp. 163:1-163:10, 2009.

- [13] Beal J., Bachrach J., "Infrastructure for engineered emergence on sensor/actuator networks," *Intell Syst IEEE*, vol. 21, pp. 10-19, 2006.
- [14] Beck L., "The therapeutic use of music for cancer-related pain," *Oncol Nurs Forum*, vol. 18, pp. 1327-37, 1991.
- [15] Bentley P.J., Corne D.W., "An introduction to creative evolutionary systems," *En P. J. Bentley and D. W. Corne, editors, Creative Evolutionary Systems*. San Francisco: Morgan Kaufmann Publishers Inc., 2001, pp. 1-75.
- [16] Bentley P., Kumar S., "Three ways to grow designs: A comparison of embryogenies for an evolutionary design problem," *En Actas del Genetic and Evolutionary Computation Conference*, 1999, pp. 35-43.
- [17] Berger K., "100 Top Stories of 2012. Digital Composer Records with London Symphony Orchestra," *Discover Magazine*, vol. 24(1), 2013.
- [18] Bergou M., Mathur S., Wardetzky M., Grinspun E., "TRACKS: Toward Directable Thin Shells," *ACM Transactions on Graphics (SIGGRAPH)*, vol. 26(3), pp. 50:1-50:10, 2007.
- [19] Bharucha J., "MUSACT: "A connectionist model of musical harmony"," en *Machine Models of Music*.: The MIT Press, 1993, pp. 497-509.
- [20] Blackwell T.M., Bentley P., "Improvised music with swarms," *En Actas del IEEE Conference on Evolutionary Computation*, 2002, pp. 1462–1467.
- [21] Boden M.A., "Computer models of creativity," *AI Magazine*, vol. 30(3), pp. 23-34, 2009.
- [22] Boden M.A., *The Creative Mind: Myths and Mechanisms*.: Basic Books, 1991.
- [23] Bongard J., "Incremental approaches to the combined evolution of a robot's body and brain," in *Doctoral dissertation, Universitat Zurich*., 2003.
- [24] Bongard J.C., Pfeifer R., "Evolving complete agents using artificial ontogeny," *En Morpho-functional Machines: The New Species*.: Springer Japan, 2003, pp. 237-258.

- [25] Bongard J., Zykov V., Lipson H., "Resilient machines through continuous self-modeling," *Science*, pp. 1118-1121, 2006.
- [26] Borenstein E., Krakauer D.C., "An end to endless forms: Epistasis, phenotype type distribution bias, and nonuniform evolution.," *PLoS Computational Biology*, vol. 4(10), 2008.
- [27] Brown C.J., Chen C., Dworkin S.F., "Music in the control of human pain," *Music Therapy*, vol. 8, pp. 47-60, 1989.
- [28] Browne C., *Automatic Generation and Evaluation of Recombination Games. Ph.D. diss., Queensland University of Technology, Australia., 2008.*
- [29] Cardle M., Barthe L., Brooks S., Robinson P., "Music-driven motion editing: local motion transformation guided by music analysis," *En Actas del Eurographics*, 2002, pp. 38-44.
- [30] Cepeda M.S., Carr D.B., Lau J., Alvarez H., "Music for pain relief," *Cochrane Database Systematic Review(2).CD004843*, 2006.
- [31] Chambel T. et al., "Creating video art with evolutionary algorithms," *Computers & Graphics*, vol. 31(6), pp. 837-847, 2007.
- [32] Chanda M.L., Levitin D.J., "The neurochemistry of music," *Trends Cogn Sci*, vol. 17, pp. 179-93, 2013.
- [33] Cheney N., MacCurdy R., Clune J., Lipson H., "Unshackling evolution: Evolving soft robots with multiple materials and a powerful generative encoding," *En Actas del Genetic and Evolutionary Computation Conference (GECCO)*, 2013, pp. 167-174.
- [34] Choudhury N., Arner I., Daniels M., and Wareing M.J., "Audiovisual distraction reduces pain perception during aural microsuction," *Annals of the Royal College of Surgeons of England*, vol. 95(1), pp. 34-6, 2013.
- [35] Cianchetti M. et al, "Soft Robotics Technologies to Address Shortcomings in Today's Minimally Invasive Surgery: The STIFF-FLOP," *Soft Robotics*, vol. 1(2), pp. 122-131, 2014.

- [36] Coca A.E., Romero R.A.F., Zhao L., "Generation of composed musical structures through recurrent neural networks based on chaotic inspiration," *En Actas del International Joint Conference on Neural Networks*, 2011, pp. 3220–3226.
- [37] Coen E., "The Art of Genes," *Oxford University Press*, 2000.
- [38] Colton S., *Automated Theory Formation in Pure Mathematics*.: Springer, 1 edition, 2002.
- [39] Cope D., "Computer modeling of musical intelligence in EMI," *Computer Music Journal*, vol. 16(2), pp. 69-83, 1992.
- [40] Cope D., "Computer Models of Musical Creativity," 2005.
- [41] Cope D., *The Algorithmic Composer*.: A-R Editions, 2000.
- [42] Corne D.W., Bentley P.J., "An introduction to creative evolutionary systems," in *Creative Evolutionary Systems*.: Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2001, pp. 1-75.
- [43] Corne D.W., Bentley P.J., *Creative Evolutionary Systems*.: Morgan Kaufmann, 2001.
- [44] Coros S., Karpathy A., Jones B., Reveret L., van de Panne M., "Locomotion skills for simulated quadrupeds," *ACM Transactions on Graphics*, vol. 30(4), p. 59, 2011.
- [45] Coumans E., "Bullet physics library," 2006.
- [46] Davidson C., "Creatures from Primordial Silicon," *New Scientist*, pp. 30-35, 1997.
- [47] De Jong K.A., "Evolutionary Computation: A Unified Approach," *The MIT Press*, 2002.
- [48] De Lasa M., Mordatch I., Hertzmann A., "Feature-Based Locomotion Controllers," *ACM Transactions on Graphics*, vol. 29(3), p. 131, 2010.
- [49] Delgado-Mata C., Ibanez J., Bee S., "On the use of Virtual Animals with



- Artificial Fear in Virtual Environments," *New Generation Computing*, vol. 25(2), pp. 145-169, 2007.
- [50] Díaz-Jerez G., "Composing with Melomics: Delving into the computational world for musical inspiration," *Leonardo Music Journal*, 2011.
- [51] Doursat R., "Organically grown architectures: creating decentralized, autonomous systems by embryomorphic engineering. In: R.P.Würtz (ed.) Organic computing, "Understanding complex systems" Series. Springer-Verlag," pp. 167–199, 2008.
- [52] Doursat R., "The growing canvas of biological development: multiscale pattern generation on an expanding lattice of gene regulatory networks," *InterJournal: Complex Systems*, p. 1809, 2006.
- [53] Doursat R., Sayama H., Michel O., *Morphogenetic Engineering: Toward Programmable Complex Systems*. "Understanding Complex Systems" Series, Springer-Verlag ed., 2012.
- [54] Doursat R., Sanchez C., "Growing fine-grained multicellular robots," *Soft Robotics*, vol. 1(2), pp. 110-121, 2014.
- [55] Doursat R., Sánchez C., Dordea R., Fourquet D., Kowaliw T., "Embryomorphic Engineering: Emergent innovation through evolutionary development," *En Morphogenetic Engineering: Toward Programmable Complex Systems*.: Springer-Verlag, 2012, pp. 275-311.
- [56] Doursat R., Sayama H., Michel O., "A review of morphogenetic engineering. "Frontiers of Natural Computing" (FNC 2012) Special Issue," *Natural Computing*, vol. 12(2), pp. 517-535, 2013.
- [57] Dubbin G.A., Stanley O., "Learning to Dance through Interactive Evolution," *En Actas del 2010 international conference on Applications of Evolutionary Computation*, vol. 2, 2010, pp. 331-340.
- [58] Ebcioglu K., "An expert system for harmonizing four-part chorales," *Computer Music Journal*, vol. 12(3), pp. 43-51, 1988.

- [59] Edwards J., "Possibilities and problems for evidence-based practice in music therapy," *The Arts in Psychotherapy*, vol. 32, pp. 293-301, 2005.
- [60] Edwards J., "Using the Evidence Based Medicine framework to support music therapy posts in healthcare settings," *British Journal of Music Therapy*, vol. 16, pp. 29-34, 2002.
- [61] Espinoza J., González L., Ruiz P., "Tratamiento de la hipersensibilidad dentinaria post terapia periodontal, mediante el uso de dos dentífricos desensibilizantes," *Rev. Clin. Periodoncia Implantol. Rehabil. Oral*, vol. 6(2), pp. 78-82, 2013.
- [62] Fernández J.D., Lobo D., Martín G.M., Doursat R., Vico F.J., "Emergent diversity in an open-ended evolving virtual community," *Artificial Life*, vol. 18(2), pp. 199-222, 2012.
- [63] Fernández J.D, Vico F.J., "AI Methods in Algorithmic Composition: A Comprehensive Survey," *Journal of Artificial Intelligence Research*, vol. 48, pp. 513-582, 2013.
- [64] Floreano D. Urzelai J., "Neural morphogenesis, synaptic plasticity and evolution," *Theory in Biosciences*, vol. 120(3-4), pp. 225-240, 2001.
- [65] Fod A., Mataric M.J, Jenkins O., "Automated derivation of primitives for movement classification," *Autonomous Robots*, vol. 12(1), pp. 39-54, 2002.
- [66] Fogel L., Owens A.J., Walsh M.J.: *Artificial Intelligence Through Simulated Evolution* John Wiley, 1966.
- [67] Foster M., "Realistic Animation of Liquids," *Graphical Models and Image Processing*, 1996.
- [68] Gabora L., DiPaola S., "Incorporating characteristics of human creativity into an evolutionary art algorithm," *En Actas del Genetic and Evolutionary Computing Conference, GECCO2007*, 2007, pp. 2442-2449.
- [69] Geis M. Middendorf M., "Creating melodies and baroque harmonies with ant colony optimization," *International Journal of Intelligent Computing and Cybernetics*,

- vol. 1(2), pp. 213–218, 2008.
- [70] Gervás P., "An expert system for the composition of formal spanish poetry," *Journal of Knowledge-Based Systems*, vol. 14(3-4), pp. 181-188, 2001.
- [71] Gill S., "A technique for the composition of music in a computer," *The Computer Journal*, vol. 6(2), pp. 129-133, 1963.
- [72] Goldman C., Gang D., Rosenschein J., Lehmann D., "NETNEG: a hybrid interactive architecture for composing polyphonic music in real time," *En Actas del International Computer Music Conference*, 1996, pp. 133-140.
- [73] Grzeszczuk R., Terzopoulos D., "Automated learning of muscle-actuated locomotion through control abstraction," *En ACM SIGGRAPH Papers Conference Proceedings*, 1995, pp. 63–70.
- [74] Grzeszczuk R. Terzopoulos D., "Automated learning of muscle-actuated locomotion through control abstraction," *En Actas del 22nd annual conference on Computer graphics and interactive techniques*, 1995, pp. 63-70.
- [75] Guangyu X., Junyun T., Dannenberg R., Veloso M., "Autonomous robot dancing driven by beats and emotions of music," *En Actas del 11th International Conference on Autonomous Agents and Multiagent Systems*, 2012, pp. 205-212.
- [76] Guétin S. et al., "The effects of music intervention in the management of chronic pain: a single-blind, randomized, controlled trial," *Clin J Pain*, vol. 28, pp. 329-37, 2012.
- [77] Guo H., Meng Y., Jin Y., "A cellular mechanism for multi-robot construction via evolutionary multi-objective optimization of a gene regulatory network," *BioSystems*, vol. 98(3), pp. 193-203, 2009.
- [78] Hahn C., Diefenbach P., "Surge: an experiment in real-time music analysis for gaming," in *ACM SIGGRAPH ASIA 2010*, 2010.
- [79] Harris R., "Algorithmic composition of jazz," en *Master's thesis*: University of Bath, 2008.
- [80] Hartling L. et al., "Music to reduce pain and distress in the pediatric emergency

- department: a randomized clinical trial," *JAMA Pediatr*, vol. 167, pp. 826-35, 2013.
- [81] Hass R., Brandes V., *Music that Works. Contributions of Biology, Neurophysiology, Psychology, Sociology, Medicine and Musicology*. Vienna: Springer, 2009.
- [82] Hauck M., Metzner S., Rohlfes F., Lorenz J., Engel A.K., "The influence of music and music therapy on pain-induced neuronal oscillations measured by magnetoencephalography," *Pain*, vol. 154, pp. 539-47, 2013.
- [83] Heard W.B., *Rigid Body Mechanics: Mathematics, Physics and Applications*.: Wiley-VCH, 2006.
- [84] Hemelrijk C.K., Hildenbrandt H., "Some causes of the variable shape of flocks of birds," *PLoS one*, vol. 6(8), 2011.
- [85] Hiller L.A., Isaacson L.M., "Musical composition with a High-Speed digital computer," *Journal of the Audio Engineering Society*, vol. 6(3), pp. 154-160, 1958.
- [86] Hodgins J., "Biped gait transitions," *En Actas del IEEE International Conference on Robotics and Automation* , 2005, pp. 2092–2097.
- [87] Hoffmann P., "Towards an "automated art": Algorithmic processes in Xenakis' compositions," *Contemporary Music Review*, vol. 21(2-3), pp. 121–131, 2002.
- [88] Holland J., *Adaptation in natural and artificial systems*.: University of Michigan Press, 1975.
- [89] Hornby G.S., Lipson H., Pollack J.B., "Generative representations for the automated design of modular physical robots," *IEEE Transactions on Robotics and Automation*, vol. 19(4), pp. 703-719, 2003.
- [90] Hornby G.S., Lohn J.D., Linden D.S., "Computer-Automated Evolution of an X-Band Antenna for NASA's Space Technology Mission," *Evolutionary Computation*, MIT Press, vol. 19(1), pp. 1-23, 2011.
- [91] Hornby G.S., Pollack J.B., "Creating high-level components with a generative representation for body-brain evolution," *Artificial Life*, vol. 8(3), pp. 152-160,

- 2013.
- [92] Huxley J., *Evolution: The Modern Synthesis.*: Allen & Unwin, London, 1942.
- [93] Jain S., Liu C., "Modal-space control for articulated characters," *ACM Transactions on Graphics*, vol. 30(5), p. 118, 2011.
- [94] Jin Y., Meng Y., "Evolutionary Developmental Robotics: The Next Step to Go?," *IEEE CIS AMD Newsletter*, vol. 8(1), pp. 13-14, 2011.
- [95] Joachimczak M., Kowaliw T., Doursat R., Wróbel B., "Evolutionary design of soft-bodied animats with decentralized control," *Artificial Life and Robotics*, vol. 18(3-4), pp. 152-160, 2013.
- [96] Johnson M., Tauritz D.R., Wilkerson R., "Evolutionary computation applied to melody generation," *En Actas del Artificial Neural Networks in Engineering (ANNIE) Conference*, 2004.
- [97] Juslin P., Liljestro S., Västfjäll D., Barradas G., Silva A., "An Experience Sampling Study of Emotional Reactions to Music: Listener, Music, and Situation," *Emotion*, vol. 8(5), pp. 668-683, 2008.
- [98] Kim T., Il Park S., Shin S.Y., "Rhythmic-motion synthesis based on motion-beat analysis," *ACM Trans. Graph*, vol. 22(3), pp. 392-401, 2003.
- [99] Kim J., Pollard N.S., "Direct control of simulated non-human characters," *IEEE Computer Graphics and Applications*, vol. 31(4), pp. 56-65, 2011.
- [100] Kim J., Pollard N.S., "Fast simulation of skeleton-driven deformable body characters," *ACM Transactions on Graphics*, vol. 30(5), 2011.
- [101] Kodjabachian J., Meyer J.A., "Development, learning and evolution in animates. From Perception to Action," *IEEE Press*, 1994.
- [102] Komosinski M., Rotaru-Varga A., "Comparison of different genotype encodings for simulated three-dimensional agents," *Artificial Life*, vol. 7(4), pp. 395-418, 2001.
- [103] Kondo R., Kanai T., Anjyo K., "Directable animation of elastic objects," *En*

- Actas del 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation*, 2005, pp. 127-134.
- [104] Koos S., Mouret J.M, Doncieux S., "The transferability approach: Crossing the reality gap in evolutionary robotics," *IEEE Transactions on Evolutionary Computation*, 2012.
- [105] Koza J.R., *Genetic programming: On the programming of computers by means of natural selection.*: The MIT Press, 1992.
- [106] Lee Y., Kim S., Lee J., "Data-driven biped control," *ACM Transactions on Graphics*, vol. 29(4), p. 129, 2010.
- [107] Lenat D., "On automated scientific theory formation: A case study using the am program," en *Machine Intelligence 9.*: Ellis Horwood, 1979, pp. 251-283.
- [108] Li Y., Dong Y., "Preoperative music intervention for patients undergoing cesarean delivery," *Int J Gynaecol Obstet*, vol. 119, pp. 81-3, 2012.
- [109] Lipson H., "Evolutionary robotics and open-ended design automation".
- [110] Lipson H., Pollack J.B., "Automatic design and manufacture of artificial lifeforms," *Nature* 406, pp. 974–978, 2000.
- [111] Lipson H., Schmidt M., "Distilling free-form natural laws from experimental data," *Science*, vol. 324(5923), pp. 81-85, 2009.
- [112] Lobo D., Fernández J.D., Vico F.J., "Behavior-Finding: Morphogenetic Designs Shaped by Function," En *Morphogenetic Engineering: Toward Programmable Complex Systems.*: Springer-Verlag, 2012, pp. 441-472.
- [113] Lobo D., Vico F.J., "Evolution of form and function in a model of differentiated multicellular organisms with gene regulatory networks," *Biosystems*, vol. 102(2), pp. 112-123, 2010.
- [114] Lobo D., Vico F.J., "Evolutionary development of tensegrity structures," *BioSystems* , vol. 101(3), pp. 112-123, 2010.

- [115] Lobo D., Vico F.J, Dawson J., "Graph grammars with string-regulated rewriting," *Theoretical Computer Science*, vol. 412(43), pp. 6101-6111, 2011.
- [116] Manaris B., Hughes D., Vassilandonakis Y., "Monterey mirror: Combining Markov models, genetic algorithms, and power laws," *En Actas del IEEE Conference on Evolutionary Computation*, 2011.
- [117] Manos S., Large M. C. J., Poladian L., "Evolutionary Design of Single-Mode Microstructured Polymer Optical Fibres using an Artificial Embryogeny Representation," *En Actas del Genetic and Evolutionary Computation Conference*, 2007, pp. 2549-2556.
- [118] Marques V.M., Oliveira V., Vieira S., Rosa A.C., "Music composition using genetic evolutionary algorithms," *En Actas del IEEE Conference on Evolutionary Computation*, 2000, pp. 714-719.
- [119] Martin S., Thomaszewski B., Grinspun E., Gross M., "Example-based elastic materials," *ACM Trans. Graph*, pp. 72:1–72:8, 2011.
- [120] Matsota P. et al., "Music's use for anesthesia and analgesia," *J Altern Complement Med*, vol. 19, pp. 298-307, 2013.
- [121] Matsushita K., Lungarella M., Paul C., Yokoi H., "Locomoting with less computation but more morphology," *En Actas del 22nd IEEE International Conference on Robotics and Automation*, 2005, pp. 2008-2013.
- [122] McIntyre R.A., "Bach in a box: the evolution of four part baroque harmony using the genetic algorithm," *En Actas del IEEE Conference on Evolutionary Computation*, 1994, pp. 852-857.
- [123] Meehan J.R., *The metanovel: writing stories by computer. PhD thesis, New Haven, CT, USA.*, 1976.
- [124] Miller G.S.P., "The motion dynamics of snakes and worms," en *SIGGRAPH Comput. Graph*, vol. 22, 1988, pp. 169-173.
- [125] Miller J.F, Banzhaf W., "Evolving the program for a cell: from french flags to boolean circuits," *On Growth, Form and Computers*, pp. 278-301, 2003.

- [126] Miranda E., "Composing Music with Computers (Music Technology)," *Focal Press*, 2001.
- [127] Mitchell M., "An Introduction to Genetic Algorithms," *The MIT Press*, 1998.
- [128] Molfino R., Armada M., Cepolina F., Zoppi M., "ROBOCLIMBER the 3 ton spider," *Industrial Robot*, vol. 32(2), pp. 163-170, 2005.
- [129] Morgan N., "Transformation and mapping of L-Systems data in the composition of a large-scale instrumental work," *En Actas del European Conference on Artificial Life*, 2007.
- [130] Müller G.B., "Evo-devo: extending the evolutionary synthesis," *Nature Reviews*, vol. 8(12), pp. 943-949, 2007.
- [131] Murata S. et al., "M-TRAN: Self-reconfigurable modular robotic system," *IEEE/ASME Transactions on Mechatronics*, vol. 7(4), pp. 431-441, 2002.
- [132] Neff M., Fiume E., "Modeling tension and relaxation for computer animation," *En Actas del 2002 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 2002, pp. 81-88.
- [133] O'Grady R., Christensen A.L, Dorigo M., "SWARMORPH: multirobot morphogenesis using directional self-assembly," *IEEE Transactions on Robotics*, vol. 25(3), pp. 738-743, 2009.
- [134] Papadopoulos G., Wiggins G., "A genetic algorithm for the generation of jazz melodies," *En Actas del Finnish Conference on Artificial Intelligence (STeP)*, 1998.
- [135] Paul C., "Morphological computation: a basis for the analysis of morphology and control requirements," *Robotics and Autonomous Systems*, vol. 54(8), pp. 619-630, 2006.
- [136] Paul C., Valero-Cuevas F.J., Lipson H., "Design and control of tensegrity robots for locomotion," *IEEE Transactions on Robotics*, vol. 22(5), pp. 944-957, 2006.
- [137] Pollack J.B., Hornby G.S, Lipson H, and Funes P., "Computer creativity in the automatic design of robots," *Leonardo, Journal for the International Society for Arts*



- Sciences and Technology*, vol. 36(2), 2003.
- [138] Prusinkiewicz P., "Score generation with L-systems," *En Actas del International Computer Music Conference*, 1986, pp. 455-457.
- [139] Prusinkiewicz P., Lindenmayer A., *The algorithmic beauty of plants.*: Springer-Verlag, 1990.
- [140] Radevsky R.A., Garas F.K., "Advanced robotics in the field of survey/inspection maintenance and repair of buildings and structures," en *5th International Symposium on robotics in construction*, 1988.
- [141] Raibert M., Hodgins J.K., "Animation of dynamic legged locomotion," *ACM SIGGRAPH Computer Graphics*, vol. 25(4), pp. 349–358, 1991.
- [142] Requena G., Sánchez, C., Corzo-Higueras, J.L., Reyes-Alvarado, S., Rivas-Ruiz, F., Vico, F., Raglio, A., "Melomics music medicine (M3) to lessen pain perception during pediatric prick test procedure," *Pediatric Allergy and Immunology*, 2014.
- [143] Reynolds C.W., "Flocks, herds and schools: A distributed behavioral model," *ACM SIGGRAPH Computer Graphics*, vol. 21(4), pp. 25-34, 1987.
- [144] Rieffel J., Knox D., Smith S., Trimmer B., "Growing and evolving soft robots," *Artificial Life*, vol. 20(1), pp. 143-162, 2014.
- [145] Rieffel J. et al., "Evolving soft robotic locomotion in PhysX," *En Actas del 11 Annual Conference Companion on Genetic and Evolutionary Computation Conference*, 2009, pp. 2499-2504.
- [146] Rieffel J., Valero-Cuevas F., Lipson H., "Automated discovery and optimization of large irregular tensegrity structures," *Computers & Structures*, vol. 87(5), pp. 368-379, 2009.
- [147] Ritchie G., "Some empirical criteria for attributing creativity to a computer program," *Minds and Machines*, vol. 17(1), pp. 67-99, 2007.
- [148] Ritchie K., Binsted G., "Computational rules for generating punning riddles," *International Journal of Humor Research*, vol. 10(1), pp. 25-76, 1997.

- [149] Rizki M., Zmuda M., Tamburino L., "Evolving Pattern Recognition Systems," *IEEE Transactions on Evolutionary Computation*, pp. 594-609, 2002.
- [150] Rossi D., Zoltán N., Schlueter A., "Soft Robotics for Architects: Integrating Soft Robotics Education in an Architectural Context," vol. 1(2), pp. 147-153, 2014.
- [151] Rubenstein M., Ahler C., Nagpal R., "Kilobot: A low cost scalable robot system for collective behaviors," En *IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 3293-3298.
- [152] Sanchez C., Moreno F., Albarracín D., Fernández J.D., Vico F.J., "Melomics: A Case-Study of AI in Spain.," *AI Magazine*, vol. 34(3), pp. 99-103, 2013.
- [153] Sauer D. Yang Y.H., "Music-driven character animation," *Transactions on Multimedia Computing, Communications, and Applications*, vol. 5(4), p. 16 pages, 2009.
- [154] Scheirer E.D., "Tempo and beat analysis of acoustic musical signals.," *Journal of Acoustical Society of America*, vol. 103(1), pp. 588-601, 1998.
- [155] Schraft R.D., Simons F., "Mechanical Design of an Autonomous, Lightweight Robot for Window Cleaning," en *International Symposium on Robotics (ISR)* , 2002.
- [156] Schramm L., Jin Y., Sendhoff B., "Emergent coupling of motor control and morphological development in evolution of multi-cellular animats," en *Advances in Artificial Life: Darwin Meets von Neumann*, 2011, pp. 27–34.
- [157] Seyama J., Nagayama R., "The Uncanny Valley: Effect of Realism on the Impression of Artificial Human Faces," *Teleoperators and Virtual Environments*, vol. 6(4), pp. 337-351, 2007.
- [158] Shiratori T., Nakazawa A., Ikeuchi K., "Dancing-to-music character animation," *Eurographics*, vol. 25(3), 2006.
- [159] Shiratori T., Nakazawa A., Ikeuchi K., "Detecting dance motion structure through music analysis," En *Actas del International Conference on Face and Gesture*

- Recognition*, 2004, pp. 857–862.
- [160] Sims K., "Evolving 3D morphology and behaviour by competition," *Artificial Life*, pp. 28-39, 2009.
- [161] Sims K., "Evolving virtual creatures," En *ACM SIGGRAPH Papers*, 1994, pp. 15-22.
- [162] Sims K., "Particle Animation and Rendering Using Data Parallel Computation," *Computer Graphics*, vol. 24(4), 1990.
- [163] Sipper M., "A phylogenetic, ontogenetic, and epigenetic view of bio-inspired hardware systems," *IEEE Trans. on Evolutionary Computation*, vol. 1(1), pp. 83-97, 1997.
- [164] Smith R., "Open Dynamics Engine User Guide v0.5," 2006.
- [165] Stanley K.O., Mikkulainen R., "A taxonomy for artificial embryogeny," *Artificial Life*, vol. 9(2), pp. 93–130, 2003.
- [166] Strapparava C., Stock O., "The act of creating humorous acronyms," *Applied Artificial Intelligence*, vol. 19(2), pp. 137-151, 2005.
- [167] Sufeng Y., "Computer-Assisted Language Learning in higher education," *IT in Medicine and Education*, vol. 1, pp. 137-141, 2011.
- [168] Supper M., "A Few Remarks on Algorithmic Composition," *Computer Music Journal*, vol. 25(1), pp. 48-53, 2001.
- [169] Takagi H., "Interactive evolutionary computation: Fusion of the capacities of ecoptimization and human evaluation," *En Actas del IEEE*, 2001, pp. 1275-1296.
- [170] Tan J., Gu Y., Turk G., Liu C., "Articulated swimming creatures," *ACM Transactions on Graphics*, vol. 30(4), p. 58, 2011.
- [171] Teuscher C., Mange D., Stauffer A., Tempesti G., "Bio-inspired computing tissues: Towards machines that evolve, grow, and learn," en *IPCAT*, 2001.

- [172] Thywissen K., "GeNotator: an environment for exploring the application of evolutionary techniques in computer-assisted composition," *Organised Sound*, vol. 4(2), pp. 127–133, 1999.
- [173] Todd P. M., "A connectionist approach to algorithmic composition," *Computer Music Journal*, vol. 13(4), pp. 27–43., 1989.
- [174] Tolley M.T. et al., "A Resilient, Untethered Soft Robot," *Soft Robotics*, pp. 213-223, 2014.
- [175] Tsukagoshi H., Hirose S., "Intermittent crawl gait for quadruped walking vehicles on rough terrain," en *First International Symposium on Climbing and Walking Robots, CLAWAR'98*, 1998.
- [176] Tuci E., Massera G., Nolfi S., "Active categorical perception of object shapes in a simulated anthropomorphic robotic arm," *IEEE Transactions on Evolutionary Computation*, vol. 14(6), pp. 885–899, 2010.
- [177] Turner S.R., "The Creative Process: A Computer Model of Storytelling and Creativity," *Psychology Press*, 1994.
- [178] Tu X., Terzopoulos D., "Artificial fishes: Physics, locomotion, perception, behavior," *En Actas del 21st annual conference on Computer graphics and interactive techniques, ACM*, 1994, pp. 43-50.
- [179] Vico F.J., Sanchez C., Albarracín D., "MELOMICS: Aportaciones de la informática y la biología a la musicoterapia receptiva," *En Sabbatella, P. (coord.), Identidad y Desarrollo Profesional del Musicoterapeuta en España. Actas del III Congreso Nacional de Musicoterapia. Cádiz 8-10 Octubre 2010. Granada: Grupo Editorial Universitario. ISBN: 978-84-9915-629-3*, 2011, pp. 83-93.
- [180] Vink A., Bruinsma M., "Evidence Based Music Therapy," *Music Therapy Today (online)*. Available from: <http://musictherapyworld.net>, vol. 4, p. 5, 2003.
- [181] Visell Y., "Spontaneous organisation, pattern models, and music," *Organised Sound*, vol. 9(02), pp. 151–165, 2004.
- [182] Wai A., Ho, Chan K., Xin Y., "A Novel Evolutionary Data Mining Algorithm with Applications to Churn Prediction," *IEEE Transactions on Evolutionary*

- Computation*, pp. 532-545, 2003.
- [183] Wang J., Fleet D., Hertzmann A., "Optimizing Walking Controllers for Uncertain Inputs and Environments," *ACM Transactions on Graphics*, vol. 29(4), p. 73, 2010.
- [184] Waterman R.N., "Technical Task 2. Industrial requirements - formulate specifications," en *2nd International Conference on Climbing and Walking Robots, CLAWAR'99*, 1999.
- [185] Watson R.A., "Compositional Evolution: The Impact of Sex, Symbiosis, and Modularity on the Gradualist Framework of Evolution," *The MIT Press*, 2006.
- [186] White T., Newer N., Luk B.L., Hazel J., "SADIE, A climbing robot for weld inspection in hazardous environments," en *First International Symposium on Climbing and Walking Robots CLAWAR'98*, 1998.
- [187] Wilhelms J., "Using dynamic analysis for realistic animation of articulated bodies," *Computer Graphics and Applications*, vol. 7(6), pp. 12-27, 1987.
- [188] Winfield A., Harper C, Bembrini J., "Towards dependable swarms and a new discipline of swarm engineering," *Swarm robotics*, pp. 126–142, 2005.
- [189] Wismeijer A.A., Vingerhoets A.J., "The use of virtual reality and audiovisual eyeglass systems as adjunct analgesic techniques: a review of the literature," *Annals of behavioral medicine : a publication of the Society of Behavioral Medicine*, vol. 30(3), pp. 268-278, 2005.
- [190] Woch A. Plamondon R., "Using the framework of the kinematic theory for definition of a movement primitive," *Motor Contr.*, vol. 18, pp. 547-557, 2004.
- [191] Young T., Griffin E., Philips E., Stanley E., "Music as distraction in a pediatric emergency department," *J Emerg Nurs*, vol. 36, pp. 472-3, 2010.
- [192] Zengin S. et al., "Effects of music therapy on pain and anxiety in patients undergoing port catheter placement procedure," *Complement Ther Med*, vol. 21, pp. 689-96, 2013.
- [193] Zhang H, Zhang J., Zong G., Wang W., Liu R., "Sky cleaner 3: a real

- pneumatic climbing robot for glass-wall cleaning," *Robotics & Automation Magazine, IEEE*, vol. 13(1), pp. 32-41, 2006.
- [194] Zikov V. Chan A., Lipson H., "Molecubes: An open-source modular robotics kit," En *IROS-2007 Self-Reconfigurable Robotics Workshop*, 2007.