

## TEMA 5 : ANIMACIÓN 3D

Podemos considerar que una animación describe el cambio de una imagen a lo largo del tiempo, con el suficiente número de fotogramas por segundo para dar un efecto de continuidad. Existen diversas técnicas que intentan conseguir este objetivo. A grandes rasgos podemos dividir las en dos, las de animación clásica y las de animación de síntesis por ordenador. Esta última, a su vez, puede basarse en una representación 2D ó 3D de los objetos (aunque el resultado final será siempre, obviamente, bidimensional).

La **animación clásica** genera la secuencia de imágenes por métodos pictóricos, lo que entendemos por una imagen “dibujada”, formada por píxeles cuya coloración se asigna manualmente o semiautomáticamente, por mecanismos sencillos guiados de forma manual (por ejemplo, sistemas de relleno automático). No emplea ningún tipo de síntesis para conseguir efectos de profundidad y perspectiva, sino que es labor de los dibujantes conseguir estas sensaciones por técnicas manuales. Las imágenes deben generarse una por una, aunque esta tarea suele distribuirse en varios niveles; separando el dibujo de momentos claves en la acción de los personajes, el dibujo de los fondos (que usualmente no cambian de un fotograma a otro) y las tareas de interpolación y coloreado de cada imagen.

La **animación de síntesis por ordenador** crea las imágenes por un proceso automático a partir de una representación de los objetos que forman parte de la escena y de su movimiento. Este modelo de los objetos puede ser bidimensional, con lo cual el resultado se parece más a la animación tradicional, o puede basarse en una representación 3D, a partir de la cual pueden aplicarse métodos realistas de sombreado, simulación física, etc.

En este curso vamos a centrarnos en las distintas técnicas de animación sintética 3D, basándonos en los conocimientos sobre representación de objetos y visualización que ya hemos desarrollado. Tenemos que describir, por tanto, qué técnicas son utilizadas para describir los cambios de todo tipo que sufre una escena tridimensional a lo largo del tiempo. Una animación de síntesis siempre funciona evaluando el estado de la escena y del observador para el instante correspondiente a cada fotograma y calculando la imagen correspondiente mediante alguno de los métodos de visualización existentes.

Si los fotogramas se van generando a partir de una especificación previa y se van almacenando en ficheros (o en un fichero único para toda la animación) o en cualquier otro soporte analógico o digital, hablaremos de una **animación fotograma a fotograma** u animación *off-line*. Esta es la única forma de conseguir animaciones finales de gran calidad, dado que el tiempo de cómputo de cada fotograma puede oscilar entre unos pocos segundos y varias horas.

Una situación diferente se produce cuando el proceso que crea la animación va mostrando los fotogramas inmediatamente después de producirlos, permitiendo al usuario responder inmediatamente y teniendo en cuenta estas respuestas en la síntesis de imágenes. En este caso, que podemos llamar **animación interactiva** o animación *on-line*, la frecuencia de presentación de las imágenes (frecuencia de refresco o *frame rate*) viene determinada por la velocidad de cómputo, que a su vez depende de la potencia del equipo y la complejidad de la escena.

Cuando las restricciones temporales de la animación interactiva son relativamente estrictas se habla de **animación en tiempo real**, concepto que desarrollaremos en el siguiente tema, y si además se desarrolla algún tipo de simulación encargada de actualizar el estado de la escena en cada momento teniendo en cuenta posiblemente las acciones del usuario del programa (por ejemplo, mover el ratón para indicar en qué dirección quiere moverse), entonces hablaremos de **simulación en tiempo real**.

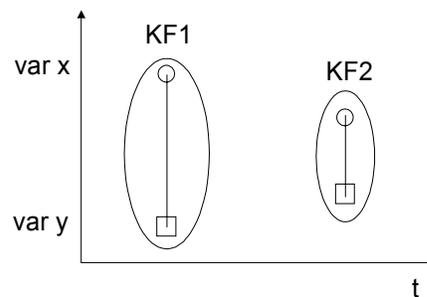
### 5.1 TÉCNICAS DE FOTOGRAMAS CLAVE O KEYFRAMES

Este tipo de técnicas es una adaptación de los métodos de la animación tradicional para describir los cambios temporales de la escena. La idea básica consiste en definir mediante edición manual o mediante alguna función automática cuál es el estado de la escena en ciertos instantes (llamados ‘fotogramas clave’ o *keyframes*). La descripción de la escena en cada fotograma clave debe incluir la posición y orientación

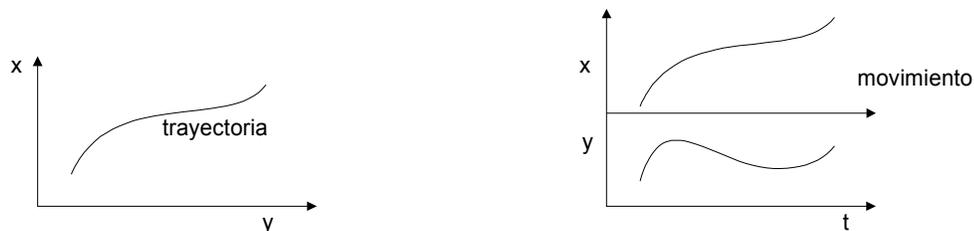
de los objetos y fuentes de luz, sus propiedades, y también la posición y características del observador o cámara. Se adapta así la idea de la animación tradicional en la que el dibujante ‘experto’ traza en blanco y negro las siluetas de los personajes en los momentos clave, para que posteriormente los interpoladores se encarguen de generar las imágenes intermedias. En la animación de síntesis la interpolación en el tiempo se efectuará a partir de los valores que determinan los fotogramas clave (posiciones, orientaciones, velocidades, propiedades...) mediante algún algoritmo automático. Este proceso de interpolación es clave, ya que debe producir un resultado coherente, de apariencia natural, sin que aparezcan saltos bruscos o cambios extraños (en algunos casos no es fácil realizar una interpolación correcta, por ejemplo en los movimientos de figuras articuladas como el cuerpo humano).

Para conseguir aproximarse más a los efectos deseados, el diseñador de la animación puede insertar mayor número de fotogramas clave (con menos separación temporal entre ellos). Además de dar los valores que definen estáticamente cada fotograma clave (p.ej. las posiciones), el diseñador puede también indicar cuál es el valor de ciertas variables dinámicas (p.ej. velocidades), lo que le otorgará un mayor grado de control sobre la interpolación.

La representación más formal de una animación por keyframes sería el llamado **diagrama de movimiento** o *motion graph*: una gráfica en la que aparecen los valores de cada una de las variables que definen la escena y su variación con el tiempo:



Asociado al diagrama básico que muestra la evolución de las variables con el tiempo podemos tener otros en los que se relacionan unas variables con otras. Por ejemplo, la trayectoria de un móvil puede verse únicamente en función de sus componentes espaciales (ver figura a la izquierda) o a través de la relación de ambas coordenadas espaciales con el tiempo (ver a la derecha):



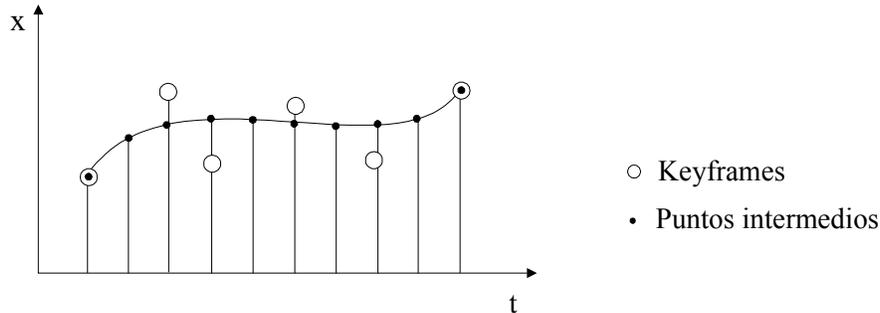
REPRESENTACIÓN DE LA TRAYECTORIA (path) + DIAGRAMA DE MOVIMIENTO  
Una variable (y) en función de la otra (x) Para añadir la componente tiempo

Podríamos pensar que a partir de la representación espacial de la trayectoria podemos averiguar la velocidad del movimiento, observando la distancia recorrida entre dos keyframes. Pero esto no es cierto, ya que el objeto no tiene por qué moverse a velocidad constante sobre la trayectoria. La velocidad instantánea real vendrá dada por las pendientes de la función de la posición en el diagrama de movimiento. Así, el vector velocidad instantáneo será  $\left(\frac{dx}{dt}, \frac{dy}{dt}\right)$ , y su módulo será la rapidez de traslación sobre la trayectoria.

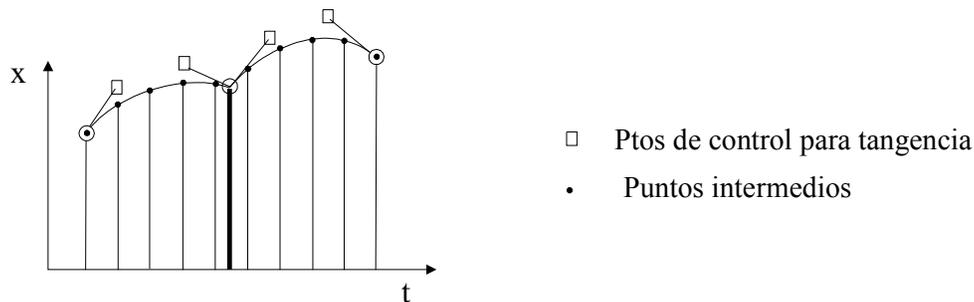
Para construir los valores intermedios de las variables a partir de los keyframes necesitamos una función de interpolación con buenas propiedades y a la vez poco costosa de evaluar. Se suelen emplear curvas paramétricas, ya que tienen un comportamiento suave y son controlables (como se comentó, cumplen la propiedad de la envolvente convexa).

Hagamos un repaso de los tipos de curvas paramétricas para comprobar cuáles pueden resultar más convenientes.

- **Splines:** La característica de las splines es que la curva resultante no tiene que pasar exactamente por los puntos de control, que en este caso son los valores de los keyframes. Esta característica hace que los keyframes dejan de representar de forma exacta la situación de la escena en ciertos instantes, lo que puede resultar inconveniente.

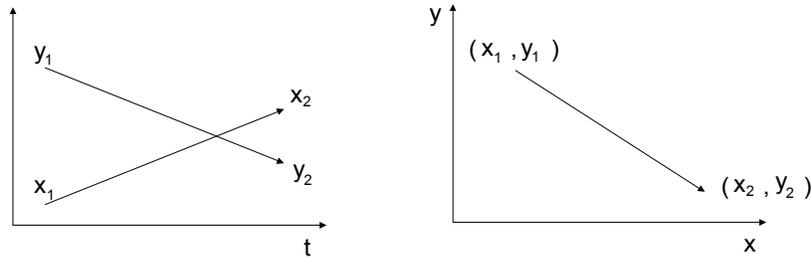


- **Curvas cúbicas de Bèzier a trozos:** Estas curvas tienen la característica de que cada trozo empieza y termina en los puntos de control, pasando por los valores correspondientes a los keyframes. Además, si añadimos dos puntos de control auxiliares en cada tramo podremos controlar la tangente de la función, y por tanto determinar de forma más exacta el camino para ir de un punto a otro. Por ejemplo, en el caso de que la variable representada sea una posición, este sistema nos permitiría controlar la velocidad del movimiento en el instante definido por el keyframe. Este tipo de curvas es, por tanto, ampliamente utilizado.

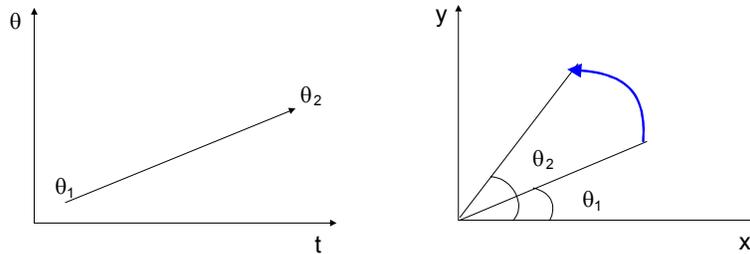


Además de estas curvas paramétricas también se puede emplear una simple interpolación lineal entre los valores de los keyframes. Esto permite resolver la interpolación mediante un cálculo muy sencillo, pero se trata de un tipo de ecuación que producirá cambios bruscos en la derivada de la variable y no permite un control detallado si no se añaden suficientes keyframes.

Además del tipo de interpolación, resulta fundamental la elección correcta de los parámetros a controlar. Por ejemplo, una interpolación lineal de la posición puede ser adecuada para una partícula que se mueve con una trayectoria suave (ver figura 5.1.(a)). Sin embargo, si deseamos describir la rotación de un objeto alrededor de un punto, resultaría poco adecuado hacerlo con una interpolación lineal de la posición, siendo mejor hacerlo con una interpolación, lineal o no, sobre un ángulo que describe el giro (ver figura 5.1.b).



Interpolación lineal de posición



Interpolación lineal del ángulo

Figuras 5.1.: (a) Interpolación de posición y (b) Interpolación de ángulo

El método de fotogramas clave deja un problema abierto: cómo realizar la **asignación de valores a las variables en los keyframes**. Podemos resolverlo de distintos modos. Si disponemos de algún método algorítmico (por ejemplo una simulación basada en ecuaciones físicas) para calcular el valor de estas variables en el tiempo, entonces podríamos usarlo para calcular el estado de la escena en ciertos instantes y luego utilizar la interpolación entre keyframes. Este sistema puede ser útil cuando el algoritmo exacto resulta demasiado costoso para utilizarlo en el cálculo de cada fotograma aislado.

En el caso de que no dispongamos de un algoritmo o procedimiento automático para calcular la evolución de las variables de la escena con el tiempo tendremos que recurrir a otros métodos.

### 5.1.1. MANIPULACIÓN DIRECTA.

La manipulación directa es el método más creativo para generar animaciones, asignando de forma manual (normalmente mediante un interfaz gráfico) los valores de las variables en cada keyframe. Todos los programas de animación 3D proporcionan interfaces de manipulación directa que permiten realizar dos tipos de tareas distintas, que pueden combinarse entre sí para garantizar la coherencia del resultado:

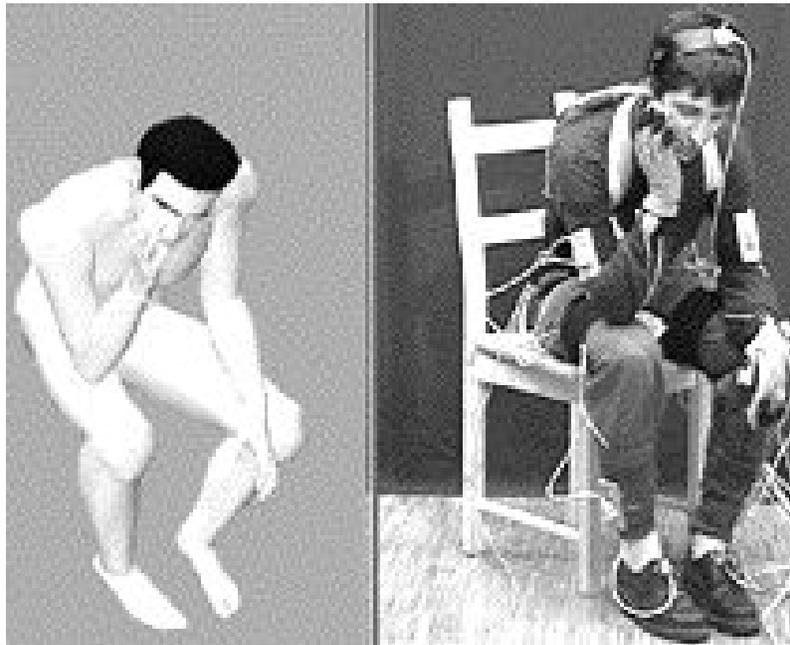
- Para determinar los keyframes, el interfaz deberá permitirnos dar la secuencia de valores para una variable dada. Para ello podremos manipular cada objeto y sus propiedades hasta colocarlo en el estado deseado y luego asignar este estado a un instante de tiempo (keyframe).
- Dada una asignación de valores a los diferentes keyframes, podremos representar la situación de la escena en cualquier instante, para comprobar si el movimiento de los distintos objetos se aproxima al efecto que queremos conseguir. Es decir, dispondremos de un modo de 'previsualización' conjunta de la escena, al menos para instantes concretos de tiempo.

A pesar de su flexibilidad y potencia expresiva, este método manual plantea el problema de la dificultad para reproducir con naturalidad comportamientos de sistemas complejos donde intervienen actores (personas, animales u objetos 'animados') o sistemas que obedecen a leyes físicas.

### 5.1.2. ROTOSCOPIA

La rotoscopia es una técnica que se basa en la grabación de los movimientos o comportamientos reales para la generación de fotogramas clave. Esta técnica se usa típicamente para reproducir la locomoción y otros movimientos de personas y animales. Tradicionalmente, la grabación se realizaba mediante fotografía o filmación y se ‘calcaba’ a mano sobre los dibujos, adaptando la forma de los personajes. Este sistema ha sido utilizado por la compañía Disney desde el primer largometraje de dibujos animados (Blancanieves).

Este método se ha extendido a la animación por ordenador mediante la *captura de movimiento*, que consiste en introducir directamente en el ordenador valores numéricos que representan, por ejemplo, las posiciones de las articulaciones en ciertos puntos seleccionados, medidas mediante dispositivos mecánicos, ópticos o electromagnéticos. De este modo se consiguen secuencias muy realistas de movimiento, que luego pueden mezclarse o deformarse en tiempo real (por ejemplo, los movimientos de los luchadores o de los jugadores de fútbol en la última generación de videojuegos).



Control en tiempo real de un actor sintético mediante captura de movimiento utilizando sensores electromagnéticos

Una vez asignados los fotogramas clave, ya hemos visto que la definición correcta de la interpolación implica el **control de la velocidad**. La velocidad, y su derivada la aceleración, son factores muy importantes del movimiento que le confieren características de más o menos realismo. Efectos como la inercia del movimiento o la acción debida a fuerzas (empujones, choques, propulsión...) involucran a la velocidad y la aceleración.

Para obtener efectos adecuados en la animación debemos especificar la velocidad y aceleración, pero teniendo en cuenta que estas magnitudes se hayan directamente ligadas al diagrama de movimiento. Si hemos definido una trayectoria  $x = f(t)$ , el diagrama de velocidad se obtendrá derivando esta función y el de aceleración derivando una vez más. A su vez, si especificamos una velocidad inicial, podemos deducir el diagrama de velocidad integrando la función de aceleración, y dando la posición inicial calcularemos la trayectoria integrando la función de velocidad, de manera que podemos crear trayectorias que cumplan con determinadas especificaciones de aceleración o velocidad.

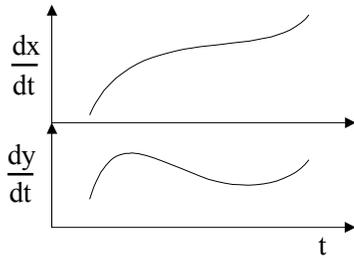


Diagrama de velocidad

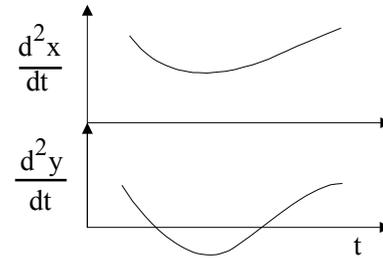


Diagrama de aceleración

## 5.2 ANIMACIÓN PROCEDURAL Y BASADA EN REGLAS

Este tipo de animaciones se caracteriza por utilizar un método algorítmico para calcular a lo largo del tiempo las variables que definen el comportamiento de los objetos, de la cámara y otras características de la escena. Estos métodos podrán utilizarse para hallar directamente el estado de la escena en cada fotograma, o bien para asignar valores a los *keyframes* y aplicar más tarde un método de interpolación. La primera variante de funcionamiento requiere que el algoritmo encargado de calcular la evolución temporal se ejecute al mismo tiempo que se va generando la animación, lo que incrementará el coste temporal total. El método de utilizar los *keyframes* como representación intermedia supone realizar una aproximación, pero como contrapartida permite ahorrar coste de computación.

En los **métodos procedurales** una función o procedimiento define la evolución del conjunto de parámetros por medio de una expresión paramétrica en función del tiempo  $X = f(t)$  o bien mediante de una expresión integral o incremental, también en función del tiempo:

$$X(t + \Delta t) = f(X(t), \Delta t).$$

En los **métodos basados en reglas** existe un sistema o conjunto de reglas que determinarán el valor de los parámetros en cada instante de tiempo. Las reglas suelen tener la forma básica *SI (condición) ENTONCES acción*, pudiendo activarse o no según la condición sea verdadera o falsa, lo que normalmente depende del valor del tiempo y de los parámetros en un instante anterior. La idea es la misma que en el caso procedural; definir cómo cambian los valores de la escena con el tiempo, pero ahora este cambio no se puede expresar mediante una función matemática. Estos métodos basados en reglas se emplean cuando aparecen discontinuidades en el tiempo (por ejemplo, un objeto se comporta de determinada manera hasta que llega cerca de otro, y entonces su movimiento se altera debido a una cierta decisión), o cuando intervienen variables externas (p.ej. interacción del usuario). Su aplicación más usual es la representación de las relaciones complejas, no debidas a meras interacciones físicas, entre los distintos objetos o actores de una escena. En algunos casos se llega a hablar de métodos basados en el conocimiento o métodos de **inteligencia artificial**.

El resultado de un sistema de reglas es difícil de evaluar a simple vista. Habrá situaciones en las que se dispare más de una regla, y por lo tanto habrá que dotar de un orden de preferencia a las reglas o especificar cómo se combinan los resultados de decisiones diferentes (esto es posible, por ejemplo, en los sistemas basados en lógica difusa o *fuzzy logic*). En algunas ocasiones las reglas pueden involucrar la eliminación o adición de nuevos elementos a la escena, lo que también puede provocar problemas de consistencia.

Existen lenguajes de animación que permiten representar ambos métodos, tanto el basado en procedimientos como en reglas, y combinarlos simultáneamente. Son los **lenguajes de script o guiones**.

Con estos métodos se pueden simular muchos sistemas naturales (nubes, fluidos, humo...) como conjuntos de *partículas* que interactúan entre sí (analizaremos esta técnica con más detalle posteriormente). Otra posibilidad consiste en el *acting*; la simulación de objetos animados complejos que pueden representarse mediante un esqueleto articulado (que puede tener 'modelos' asociados) que sigue ciertas reglas físicas y conductuales en su movimiento.

Veamos ahora un ejemplo muy sencillo de animación procedural. Se trata de un móvil que se desplaza en una dimensión  $x$  a una velocidad constante de diez unidades. Las versiones paramétrica e incremental de un procedimiento que devolviera la posición para cada instante serían :

Procedimientos

```
const v = 10;
proc_param( t ): real
{ return ( v * t ); }
```

```
const v = 10;
proc_increm( x_anterior, Δt ): real
{ return ( x_anterior + v * Δt ); }
```

Para representar la persecución entre un depredador y su presa en una dimensión podríamos definir el siguiente modelo basado en la combinación de reglas y cálculos procedurales:

Reglas

```
Variables del Depredador : Xd, Vd
Variables de la Presa : Xp, Vp
Procedimiento actualizar_pos_depredador ( Δt )
{
si | Xd - Xp | < 10 (si la presa se acerca a menos de diez unidades)
    Vd = ( Vd + Vp ) / 2 se le asigna una velocidad entre la suya y la de la presa
si | Xd - Xp | >= 10 (si la presa está más lejos)
    Vd = 0 el depredador está parado
si | Xd - Xp | < 0.001 ( si se ha cazado la presa)
    Vd = 0; Vp = 0; ambos se paran
Xp = Xp + Vp * Δt se actualiza la posición del depredador
}
```

La función definida para el depredador incrementa automáticamente su posición al moverse la presa. Una función similar podría ser utilizada para el movimiento de ésta.

**Ejemplo:**

En la siguiente figura podemos observar un ejemplo más elaborado, en el que el movimiento de un pájaro es controlado mediante una serie de variables cuyos valores puede cambiar el usuario.

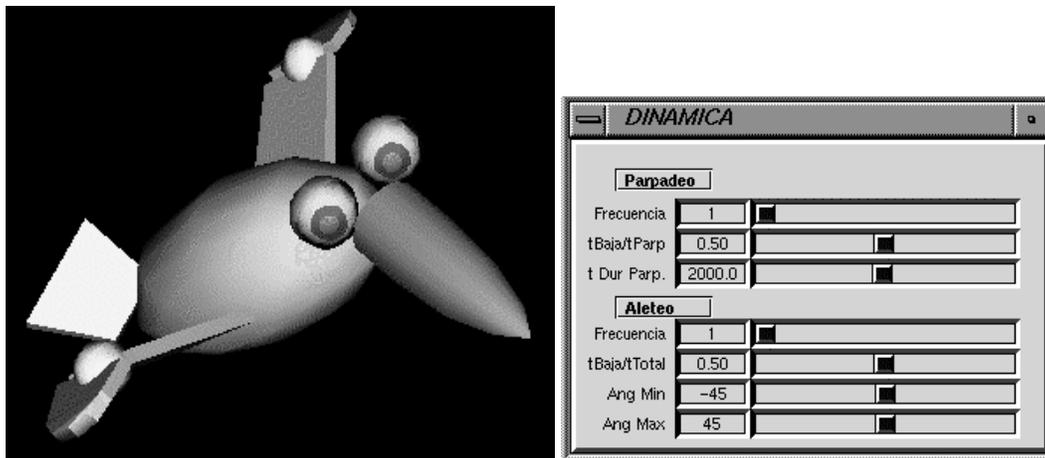


Figura 5.2.: Panel de control de parámetros para la animación procedural de un actor sintético

**5.3 ANIMACIÓN FÍSICA**

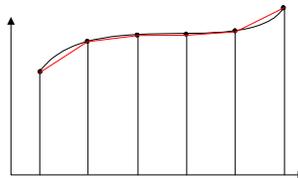
Este tipo de animación procedural utiliza un modelo matemático para describir el movimiento y otros cambios. Se supone que este modelo reproduce con cierta fidelidad la realidad física. Un modelo *cinético* solamente incluye aceleraciones y velocidades, pero no las fuerzas que se relacionan con el movimiento, mientras que un modelo *dinámico* considera estas fuerzas para deducir el valor de los parámetros cinemáticos. Las técnicas de animación o simulación física pueden implicar un coste más o menos grande según la complejidad del modelo matemático que debe evaluarse y el número de interacciones a considerar. Los casos más elementales son los siguientes.

### 5.3.1. CINEMÁTICA DEL PUNTO

En este caso la variable representativa del sistema es la posición de un punto sin dimensiones (por tanto no tiene sentido considerar la orientación y el comportamiento de rotación, el cuerpo no tiene momento de inercia ni momento angular). La masa se supone concentrada en ese único punto. Para generar la animación hay que calcular  $\mathbf{x}(t)$ , normalmente en función de la velocidad  $\mathbf{v}(t)$  y la aceleración  $\mathbf{a}(t)$ . El movimiento vendrá descrito por ecuaciones diferenciales que relacionarán estas tres funciones. La resolución se efectuará los métodos de integración apropiados.

Consideremos un caso muy sencillo en el que disponemos como dato de la función de aceleración  $\mathbf{a}(t)$ . A partir de ella podemos calcular la velocidad y la posición, utilizando, por ejemplo, el método de integración de los trapecios.

$$\left. \begin{array}{l} \bar{\mathbf{a}} = f(t) \\ \bar{\mathbf{x}}(0) = \bar{\mathbf{x}}_0 \\ \bar{\mathbf{v}}(0) = \bar{\mathbf{v}}_0 \end{array} \right\} \begin{array}{l} \text{para } t=0 \text{ hasta } t_f \text{ con paso } dt \\ \bar{\mathbf{a}}_B = f(t); \bar{\mathbf{v}}_B = \bar{\mathbf{v}}_B + (\bar{\mathbf{a}}_A + \bar{\mathbf{a}}_B)/2 \cdot dt \\ \bar{\mathbf{x}}_t = (\bar{\mathbf{v}}_A + \bar{\mathbf{v}}_B)/2 \cdot dt \\ \bar{\mathbf{v}}_A = \bar{\mathbf{v}}_B ; \bar{\mathbf{a}}_A = \bar{\mathbf{a}}_B \text{ /* pasa al siguiente trozo */} \end{array}$$



### 5.3.2. DINÁMICA DEL PUNTO

Es similar al modelo anterior, pero ahora debemos tener en cuenta las fuerzas y la masa del punto en las ecuaciones diferenciales que ligan posición, velocidad y aceleración. Según el tipo de fuerzas incluidas en el modelo deberán definir otras propiedades del objeto, como carga eléctrica, elasticidad, etc.

La ecuación básica es  $\vec{F}(t) = m \cdot \vec{a}(t)$ , que indica que a partir de la expresión de las fuerzas puede calcularse la aceleración del punto, y por tanto la velocidad y la posición. La fuerza puede depender a su vez de la posición (por ejemplo, las fuerzas elásticas) o la velocidad (fuerzas de rozamiento, fuerzas magnéticas sobre cargas eléctricas). También podemos considerar en este modelo **fuerzas impulsivas**, debidas a choques o contactos temporales con otros cuerpos que pueden modificar bruscamente la velocidad y dirección de movimiento del punto. Reorganizando la fórmula anterior podemos deducir el cambio de velocidad producido por una fuerza que actúa durante un tiempo  $dt$ :

$$\vec{F}(t) = m \cdot \vec{a}(t) = m \cdot \frac{d\vec{v}}{dt} \Rightarrow \vec{F}(t) \cdot dt = m \cdot d\vec{v}$$

Las fuerzas no impulsivas suelen considerarse generadas por un **campo de fuerzas** que actúa sobre alguna magnitud activa (masa, carga) del objeto, por ejemplo el campo gravitatorio generado por otros cuerpos.

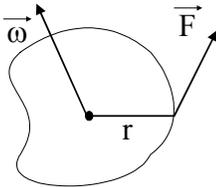
### 5.3.3. CINEMÁTICA DEL SÓLIDO RÍGIDO

La diferencia entre el modelo de sólido rígido y el del punto material es la introducción de la extensión del cuerpo, y por tanto de un nuevo comportamiento: la rotación alrededor de su centro de masas. Al movimiento propio del centro de masas pueden aplicarse las mismas ecuaciones que del modelo puntual. Pero además hay que describir la rotación a su alrededor. Ésta puede ser muy compleja, puesto que existen tres grados de libertad, y en principio habría que describir la evolución de tres ángulos de giro independientes. Sin embargo, en muchos casos podemos reducir el giro a un movimiento alrededor de un eje, y por tanto descrito por un único ángulo. El ángulo juega el mismo papel que la posición en las ecuaciones cinemáticas, donde no aparecen fuerzas. Denominando  $\theta$  al ángulo de giro tenemos:

$$\text{Velocidad angular: } \vec{\omega} = d\theta/dt \quad \text{Aceleración angular: } \vec{\alpha} = d\vec{\omega}/dt$$

### 5.3.4. DINÁMICA DEL SÓLIDO RÍGIDO

Las fuerzas actúan sobre el centro de masas del sólido rígido de la misma forma que sobre el punto material, pero las fuerzas impulsivas (y otras, como las magnéticas) influyen en el posible giro del objeto. En la dinámica de rotación, el momento de inercia (que, en principio, es una matriz 3x3) hace el papel equivalente a la masa, y el momento de una fuerza sustituye a las fuerzas en las ecuaciones de rotación. El momento de una fuerza se define como el producto vectorial entre la fuerza aplicada al objeto y el vector de posición del punto en que se aplica ( $\vec{r}$ ).



$$\vec{M} = \vec{F} \times \vec{r} \quad \vec{M} \leftarrow \text{momento de fuerza}$$

$$\vec{M} = I \cdot \vec{\alpha} \Rightarrow \vec{\alpha} = I^{-1} \cdot \vec{M} \quad I \leftarrow \text{momento de inercia}$$

Si el giro se produce alrededor de un eje fijo,  $I$  se convierte en un escalar, obteniendo:  $\alpha = \frac{M}{I}$

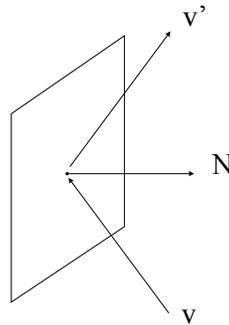
- **Modelos para colisiones (choques)**

En una colisión *elástica* en la que no actúan fuerzas externas, además de conservarse la cantidad de movimiento total (masa por vector velocidad) también se conserva la energía cinética total  $E_c = \frac{1}{2}mv^2$ .

De esta manera, en un sistema donde solamente hay colisiones elásticas no se pierde energía y los objetos siguen moviéndose indefinidamente.

En una colisión *inelástica* sí existe pérdida de energía cinética, que es absorbida por los cuerpos que chocan. Las colisiones reales siempre suponen la pérdida de alguna energía.

Resulta complicado desarrollar un buen modelo de colisión, ya que éste implica calcular con precisión el punto de contacto entre los cuerpos y resolver las condiciones impuestas por los principios de conservación y la posible pérdida de energía (incluyendo en el caso real las deformaciones que pueden sufrir los cuerpos). Si suponemos un comportamiento elástico y un punto de contacto en el que uno de los objetos puede aproximarse como una superficie localmente plana y el otro como un punto, podemos desarrollar un modelo sencillo (que serviría, por ejemplo, para representar el rebote de una pelota contra una pared o suelo).



$$\left. \begin{array}{l} \vec{N} : \text{vector normal del plano} \\ \vec{v} : \text{direccion inicial de la pelota} \\ \vec{v}' : \text{direccion resultante de la pelota} \end{array} \right\} \vec{v}' = \vec{v} - (2 \cdot \vec{v} \cdot \vec{N}) \cdot \vec{N}$$

Existen sistemas donde la interacción es mucho más compleja y no puede reducirse a una colisión de tipo clásico. Un ejemplo es el comportamiento de una tela que cubre a un cuerpo humano. Su movimiento depende de la fuerza de la gravedad, del contacto con la forma geométrica del cuerpo y de las características elásticas de la superficie de la tela, y se requiere el desarrollo de modelos matemáticos especiales.

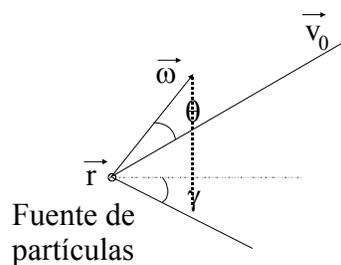
### 5.3.5. SISTEMAS DE PARTÍCULAS

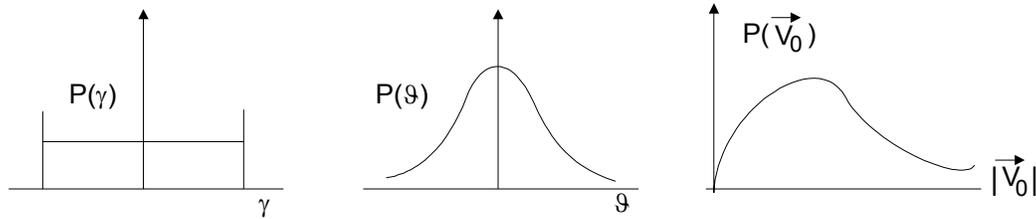
Independientemente del modelo físico que utilicemos para representar los objetos individuales, podemos considerar cómo aproximarnos al comportamiento físico de aquellos sistemas que están formados por un conjunto de cuerpos (por ejemplo, los planetas del Sistema Solar). Al igual que en los modelos de objetos aislados que hemos visto, deberemos llegar a un compromiso entre la fidelidad de la representación y el coste computacional, recurriendo a diferentes tipos de simplificaciones.

- **Sistemas de partículas independientes**

Esta sería la aproximación más sencilla y menos costosa, en la que consideramos que los diferentes objetos no se relacionan entre sí, o que lo hacen solamente en momentos determinados. Incluso con un método tan sencillo es posible representar de forma realista fenómenos complejos, como ciertos procesos naturales (gases, humo, ciertos flujos de líquidos).

Por ejemplo, si queremos simular una fuente de agua podemos suponer que cada partícula se comporta como un punto material, y generar un gran número de éstas de forma aleatoria a partir de un punto. Para realizar la generación aleatoria tendríamos que caracterizar la fuente mediante varias distribuciones estadísticas que nos den la probabilidad de los ángulos de salida y la velocidad inicial a partir del punto:  $P(\gamma)$ ,  $P(\theta)$ ,  $P(v_0)$





Es fácil comprender que en los sistemas de partículas independientes el coste computacional sería de orden lineal con el número de partículas, puesto que el cálculo del modelo individual se repite para cada uno de los componentes.

• **Sistemas de partículas con ligaduras flexibles**

Diremos que dos partículas están relacionadas mediante una ligadura flexible cuando existe una fuerza entre ellas, pero no hay una restricción en los grados de libertad de la posición de ambas. Veremos más adelante que existen otro tipo de ligaduras no flexibles, en las que los cuerpos pueden estar unidos mediante articulaciones, apareciendo restricciones en el movimiento.

**a) Inestructuradas :**

Cada una de las partículas interactúa con todas las demás. Esto sucede con la fuerza gravitatoria y otras fuerzas de largo alcance. En una simulación del Sistema Solar tendríamos en principio que considerar las fuerzas que actúan entre cada par de cuerpos  $i, j$ :

$$\vec{F}_{ij} = G \cdot \frac{m_i \cdot m_j}{r_{ij}^3} \cdot |\vec{r}_{ij}|$$

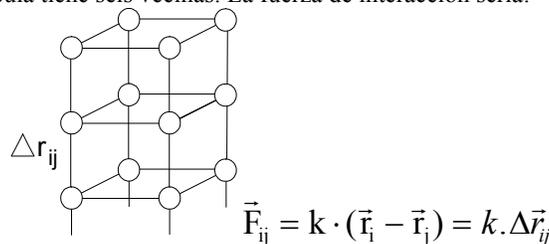
El problema de considerar todas estas interacciones es que el número de cálculos a realizar tiene un coste cuadrático con el número de partículas.

**b) Estructuradas :**

En este caso se supone que las partículas solamente interactúan con algunas de sus vecinas. Este tipo de interacción aparece, por ejemplo, cuando las partículas forman parte de una estructura que limita el alcance de las fuerzas, como sucede entre las moléculas de un líquido o un sólido. Incluso puede utilizarse esta aproximación para simular sistemas en los que la vecindad entre partículas puede cambiar (por ejemplo, en un fluido muy viscoso).

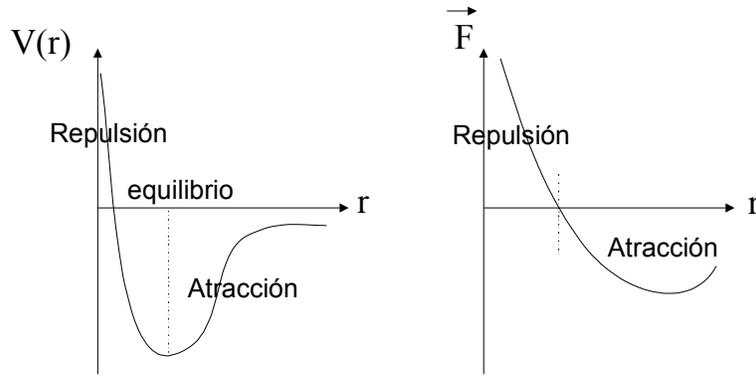
Si suponemos que el número de vecinas a considerar es más o menos constante ( $k$ ), el coste de cálculo de las interacciones sería de nuevo lineal con el número de partículas.

**Ejemplo :** Partículas en una estructura rígida con interacciones elásticas locales. En una estructura cúbica cada partícula tiene seis vecinas. La fuerza de interacción sería:



**Ejemplo :** Estructura molecular no rígida con potencial atractivo-repulsivo. Un ejemplo de este tipo serían los sistemas como geles, espuma, gelatina, líquidos viscosos, en los que estas fuerzas van haciendo evolucionar la distribución de las partículas con el tiempo. El conjunto de partículas vecinas deberá, por tanto, actualizarse cada cierto tiempo.

La relación entre la fuerza  $F$  y el potencial  $V$  es:  $\vec{F} = -\vec{\nabla}V$ . Un ejemplo de la variación de estas magnitudes según la distancia entre las partículas podría ser:



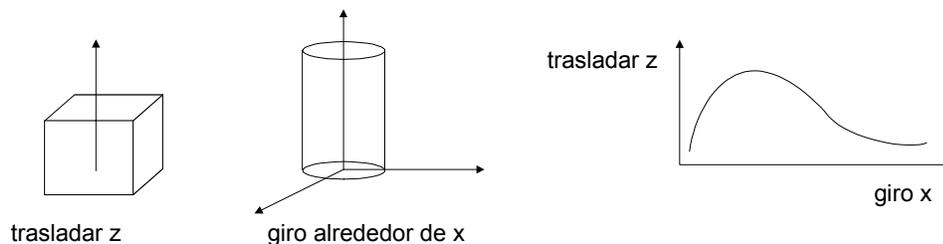
## 5.4 LIGADURAS RESTRINGIDAS Y FIGURAS ARTICULADAS

Un tipo de ligadura diferente al visto en el apartado anterior aparece cuando dos objetos se relacionan por medio de fuerzas que limitan sus movimientos, creándose una unión o contacto de alguna clase. La llamaremos ligadura no flexible, ligadura con restricciones o ligadura rígida.

El número de **grados de libertad** es la cantidad mínima de variables necesarias para especificar exactamente la posición de un objeto. Si un objeto no está sometido a ningún tipo de restricción, tiene seis grados de libertad: tres coordenadas espaciales para determinar su localización, y tres ángulos para determinar su orientación. El número de grados de libertad no varía aunque cambiemos el tipo de sistema de coordenadas (las tres coordenadas espaciales pueden ser cartesianas, esféricas o de cualquier otro tipo, pero siempre harán falta tres).

Cuando aparece una ligadura rígida el número de grados de libertad disminuye. Por ejemplo, si sujetamos el objeto a una bisagra estática eliminamos todos los grados de libertad de posición y solamente dejamos un grado de libertad de rotación.

En muchos programas de animación y simulación existe la posibilidad de establecer este tipo de ligaduras entre diferentes objetos y aprovecharlas para producir animaciones. Para establecer una ligadura rígida se suelen definir **funciones de transferencia** entre las variables de posición u orientación de diferentes objetos. Por ejemplo, podemos especificar que la posición en  $x$  de un objeto se relacione con el ángulo de giro  $\theta$  de otro objeto por medio de la función  $\theta = x/2\pi$ . Esta función debe actuar haciendo que si en cualquier momento cambia el valor de  $x$ , el valor de  $\theta$  variará según esta relación.



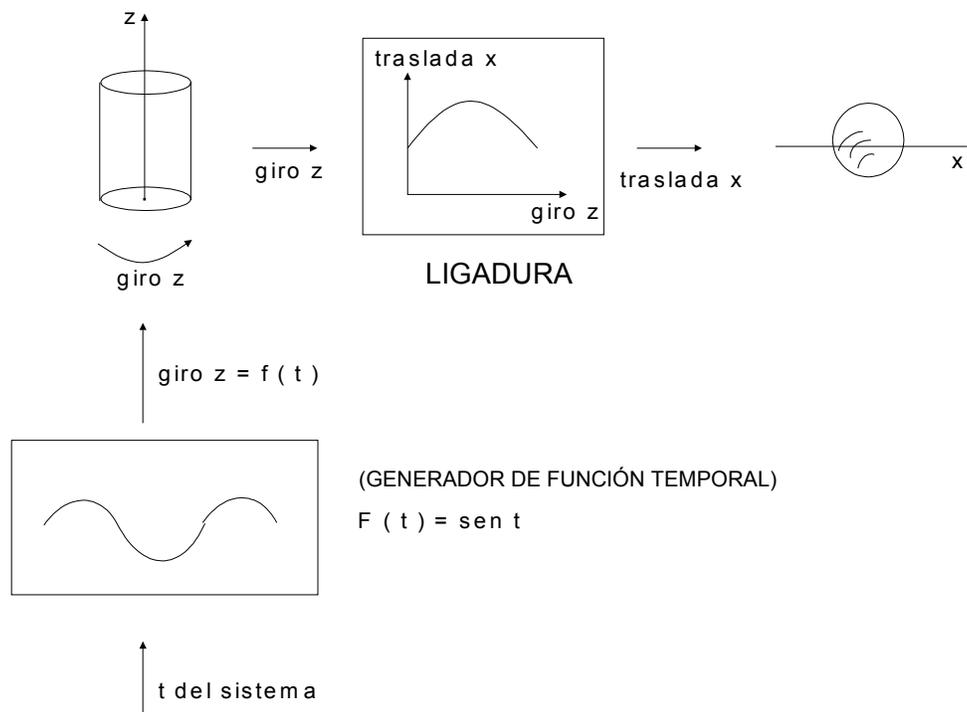
En este ejemplo, a partir del giro del cilindro se determinaría mediante una función el valor de la translación en  $z$  del cubo.

Las ligaduras establecidas mediante una función de transferencia serán reversibles cuando la función sea estrictamente creciente o decreciente, existiendo una correspondencia biunívoca entre los dos grados de libertad ligados. En nuestro ejemplo significaría que a partir de *trasladar z* se pueda obtener *giro x* y también a la inversa. Pero también pueden ser no reversibles, en el caso de que la función no sea siempre creciente o decreciente, como sucede en la gráfica del ejemplo, ya que no podemos hallar la relación inversa; a un determinado valor de 'trasladar z' podrían corresponder dos valores diferentes de 'giro x'.

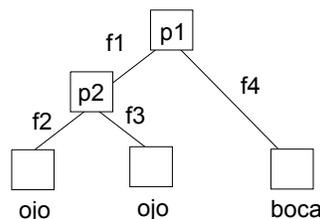
Pero establecer una relación de ligadura no especifica los valores absolutos que ambas variables van a adquirir con el tiempo. Por esta razón a las funciones de transferencia se suelen añadir los llamados **generadores de funciones temporales**, que podemos imaginar como procedimientos cuya salida es un valor que cambia en función del tiempo (por ejemplo, siguiendo una función lineal o sinusoidal). En algunos entornos para el desarrollo de animaciones se puede trabajar con una representación gráfica de las funciones de transferencia y de los generadores temporales como bloques que se conectan entre sí y también con los grados de libertad de los objetos (como muestra la siguiente figura). En otros sistemas no existe un entorno gráfico, pero sí las estructuras y funciones necesarias para crear estos objetos (como en el entorno de desarrollo dVise de Division Ltd. o en la librería gráfica IRIS Inventor).

Otra técnica utilizada para establecer ligaduras rígidas es definir **parámetros virtuales**. En lugar de establecer una correspondencia entre dos variables de objetos de la escena, como se hace en una función de transferencia, se crea un nuevo parámetro que no corresponde a ningún objeto particular pero que sirve para controlar a dos o más grados de libertad de los objetos. Esta es una manera sencilla de coordinar los movimientos de diferentes partes de un objeto o de relacionar un objeto con otro. Las ligaduras por medio de parámetros virtuales pueden anidarse, creándose una jerarquía de ligaduras, de manera que con pocas variables pueden controlarse un gran número de grados de libertad de la escena.

En la figura de ejemplo se han definido dos parámetros virtuales (p1 y p2) que permiten, a través de cuatro funciones, controlar de forma coordinada las posiciones de los dos ojos y la boca mediante el valor de p1.



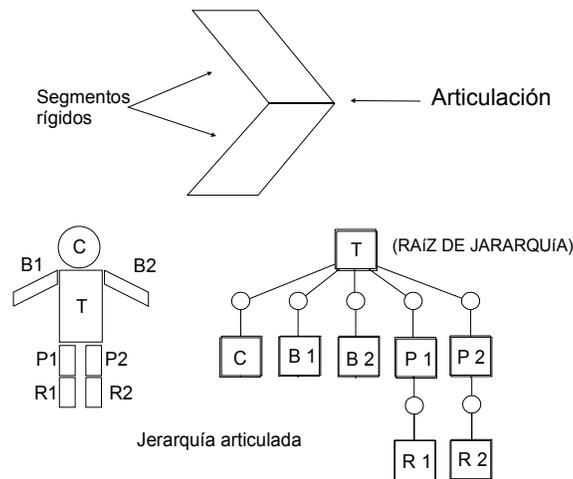
*Ejemplo de conexión entre funciones de transferencia, generadores temporales y parámetros de los objetos*



## Ejemplo de ligaduras a través de parámetros virtuales anidados

## 5.4.1. JERARQUÍAS ARTICULADAS

Una jerarquía articulada es un conjunto de partes rígidas (segmentos) que están unidas dos a dos por medio de ligaduras restrictivas llamadas **articulaciones**. Este tipo de sistemas (que la Robótica estudia en profundidad) normalmente no tiene grados de libertad relacionados con la posición, sino solamente con las rotaciones (desde uno hasta tres ángulos independientes dependiendo del tipo de articulación). Sin embargo, puesto que los objetos visuales no tienen necesariamente por qué seguir las leyes de las articulaciones reales, es posible añadir operaciones de traslación y escalado (cambio de tamaño) en cada articulación, siendo posible teóricamente tener nueve grados de libertad por cada una.



Desde el punto de vista de la informática gráfica podemos imaginar la jerarquía articulada como una estructura de datos que nos permite dibujar las partes del objeto en la posición adecuada y controlar su movimiento. El algoritmo que dibuja la escena recorre recursivamente la estructura articulada transformando el sistema de coordenadas utilizado para la representación de los objetos. Si suponemos que la representación visual de cada segmento rígido está definida respecto a un sistema de coordenadas centrado en el punto de articulación, entonces cada vez que el algoritmo de recorrido llega a una articulación de la jerarquía debe:

- 1.- Trasladar el sistema de coordenadas desde su posición anterior hasta el punto donde se conectará el siguiente segmento.
- 2.- Una vez trasladado allí, tiene que girar el sistema de coordenadas según el estado de la articulación, es decir, según los valores de los ángulos.

Si consideramos que cada articulación viene definida por tres grados de libertad de rotación, entonces son necesarios  $3*n$  valores ( $n$  es el número de articulaciones) para especificar completamente la posición de la figura articulada. El conjunto de estos  $3*n$  ángulos se llama **vector de configuración** ( $\vec{\theta}$ ).

## 5.4.2. CONTROL DE LA JERARQUÍA ARTICULADA

El problema que plantean los objetos articulados es el gran número de grados de libertad que poseen, cuyo cambio en el tiempo debe especificarse para realizar la animación. Vamos a ver cuáles son los métodos de que disponemos para conseguir que la figura realice los movimientos deseados.

Las dos formas de especificar la posición son dar el vector de configuración  $\vec{\theta}$  (los ángulos de las articulaciones), o bien dar directamente la posición final de cada segmento rígido, datos que recogemos dentro de un vector  $\vec{X}$ . Hay dos variantes básicas para hacer el control de posición con el tiempo:

En los métodos directos, dados los valores de  $\vec{\theta}$  calcularemos los de  $\vec{X}$ .

En los métodos inversos, dado  $\vec{X}$  calcularemos  $\vec{\theta}$ .

Ambos tipos de métodos pueden combinarse con keyframes de dos maneras diferentes: por un lado los keyframes pueden describir la variación con el tiempo de los datos de control (ángulos en el caso directo y las posiciones en el caso inverso); por otro lado, una vez calculado el resultado para ciertas posiciones de tiempo, estos valores pueden almacenarse en forma de keyframes. De esta última forma, cuando la animación deba ejecutarse de nuevo no será necesario calcular otra vez el movimiento de la estructura articulada, sino que se puede interpolar sobre los keyframes precalculados. En este caso hay que evitar que los keyframes se encuentren excesivamente separados en el tiempo, ya que la interpolación podría producir resultados irreales.

#### 5.4.2.1. Cinemática Directa

Es el método más sencillo. Se calcula  $\vec{X} = f(\vec{\theta})$  mediante un mero cálculo geométrico, sin tener en cuenta las propiedades dinámicas de la jerarquía articulada (inercia, resistencia, peso). Este método es muy eficiente, puesto que a partir de los ángulos de las articulaciones podemos asignar fácilmente posiciones a los segmentos recorriendo la estructura arbórea de forma recursiva tal como hemos visto antes. El inconveniente de este método es que no facilita la colocación de la figura en la postura deseada cuando existen muchos grados de libertad. Se emplea principalmente para describir movimientos que no están orientados a objetivos, sino que se desarrollan de forma mecánica y autónoma.

#### 5.4.2.2. Dinámica Directa

Al no considerar las variables dinámicas, la cinemática directa puede producir efectos irreales desde el punto de vista físico (por ejemplo, cambios instantáneos en el movimiento de un brazo articulado, sin inercia ni aparente esfuerzo). Por esta razón es conveniente hacer el cálculo directo utilizando ecuaciones diferenciales en las que aparezcan las fuerzas, momentos, torsiones, etc., aunque esto significa aumentar considerablemente el coste computacional.

#### 5.4.2.3. Cinemática Inversa

Los métodos inversos en los que se pretende llegar a una relación  $\vec{\theta} = f^{-1}(\vec{x})$  conllevan un proceso de cálculo mucho más costoso que los directos, y por tanto resultan de difícil aplicación para tiempo real, sobre todo cuando el número de grados de libertad es elevado (existen algunas aproximaciones de tipo iterativo que pueden utilizarse). Sin embargo, este tipo de métodos resultan casi imprescindibles para crear animaciones realistas de figuras articuladas cuyo comportamiento está orientado a objetivos, es decir, que tratan de alcanzar ciertas posiciones (como coger un objeto con una extremidad o caminar sobre una superficie).

Uno de los inconvenientes de los métodos inversos, y en particular de la cinemática inversa, es que pueden existir infinitas soluciones para los valores de los ángulos que llevan a una posición final dada, y debe elegirse solamente una de ellas. También es posible que, debido a las restricciones de la estructura (por ejemplo, no puede intersectarse consigo misma), no exista solución.

#### 5.4.2.4. Dinámica Inversa

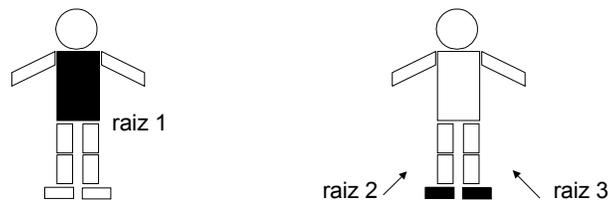
Este último método, aunque el más costoso, presenta una serie de ventajas frente a los descritos anteriormente. Por un lado la introducción de criterios dinámicos, principalmente el criterio de utilizar la mínima energía posible en el desplazamiento, permiten reducir el espacio de posibles soluciones para encontrar la óptima. Además producen resultados muy naturales, puesto que las estructuras articuladas naturales como las de los animales siguen el mismo principio de mínima energía.

#### 5.4.2.5. Otros Métodos Complementarios de Control

- **Control por puntos fijos**

Hasta ahora hemos supuesto que el nodo raíz de la jerarquía articulada era siempre el mismo. Resulta evidente que la posición del segmento rígido que actúa como raíz de la jerarquía no va a cambiar en el proceso recursivo de evaluación, ya que éste comienza precisamente a partir de ese nodo del árbol. Por tanto, la posición del segmento raíz en el espacio puede ser asignada de forma independiente y nos sirve para mover la totalidad de la figura articulada por el espacio.

Sin embargo, hay ocasiones en que nos interesaría poder determinar directamente la posición de otro segmento de la jerarquía y mantenerlo fijo durante el movimiento. Por ejemplo, cuando andamos uno de los pies se mantiene en contacto con el suelo mientras nos desplazamos, y luego cambiamos el apoyo al otro pie. Para reproducir ese comportamiento podemos comenzar por asignar el nodo que representa al pie fijo como nodo raíz, a partir del cual se calcula la posición de los otros segmentos. Podemos luego cambiar el nodo raíz de un pie a otro y reordenar la jerarquía, teniendo en cuenta que las transformaciones realizadas en cada articulación se invertirán.



Movemos el cuerpo

Movemos los pies

Cambio del nodo raíz del tronco (raiz 1) a los pies (raiz 2 y raiz 3) para controlar el movimiento al andar.

- **Control del centro de masa**

Para crear posturas estáticas y movimientos de “balanceo” realista debemos de tener en cuenta que el centro de masa de toda la figura articulada que esté en equilibrio debe situarse sobre la vertical de la superficie de sustentación (por ejemplo la formada por la planta de un pie en el caso de sostenerse sobre uno solo, o la que uniría los dos pies en el caso de apoyarse sobre ambos). En algunos movimientos, por ejemplo al andar, el centro de masa sale fuera de esta superficie temporalmente (al cambiar de un pie a otro), cuando la figura deja de estar en equilibrio.

- **Ligaduras restrictivas adicionales**

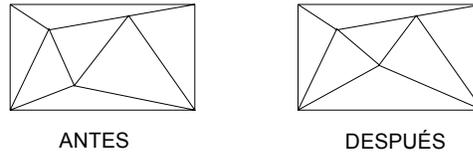
Podemos combinar la estructura articulada ligaduras entre los valores de los parámetros de las articulaciones. De esta manera disminuimos el número de grados de libertad y resulta más fácil conseguir movimientos en los que diferentes partes de la estructura están coordinadas. Por ejemplo, podemos conseguir que los dos brazos de una figura humana se muevan a la vez, ligando sus ángulos mediante funciones de transferencia que los igualan.

## 5.5 CAMBIOS DE FORMA

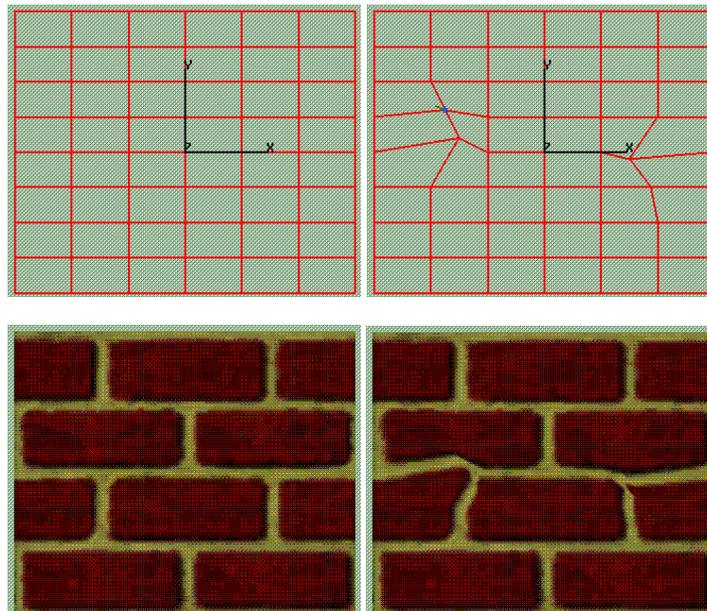
Hasta ahora hemos supuesto que los objetos no cambiaban de forma con el tiempo, pero eso no siempre se corresponde con la realidad. En muchos casos puede resultar interesante describir estos cambios de forma, bien para introducirlos en la animación o bien como una ayuda para modelar los objetos durante el proceso previo de edición hasta que adopten la forma que nos interesa. Veremos primero las técnicas utilizadas para manipular la forma en imágenes planas y luego las extenderemos a los objetos tridimensionales.

### 5.5.1 MORPHING 2D

Recientemente se han popularizado en publicidad y efectos especiales cinematográficos las técnicas llamadas de *morphing*. Debemos distinguir claramente entre aquellos efectos que se basan en la manipulación de imágenes planas de aquellos otros que trabajan con una representación tridimensional. En el *morphing* de imágenes se utilizan dos efectos básicos; la deformación de la imagen, redistribuyendo sus colores y formas, y el fundido de dos imágenes, pasando de forma continua de una a otra. Ambos efectos parten de una descomposición del espacio de la imagen en una mallado de triángulos. Para especificar un cambio continuo en la forma de la imagen basta con describir cómo varía la posición de los vértices de la malla de un instante de tiempo al instante posterior.



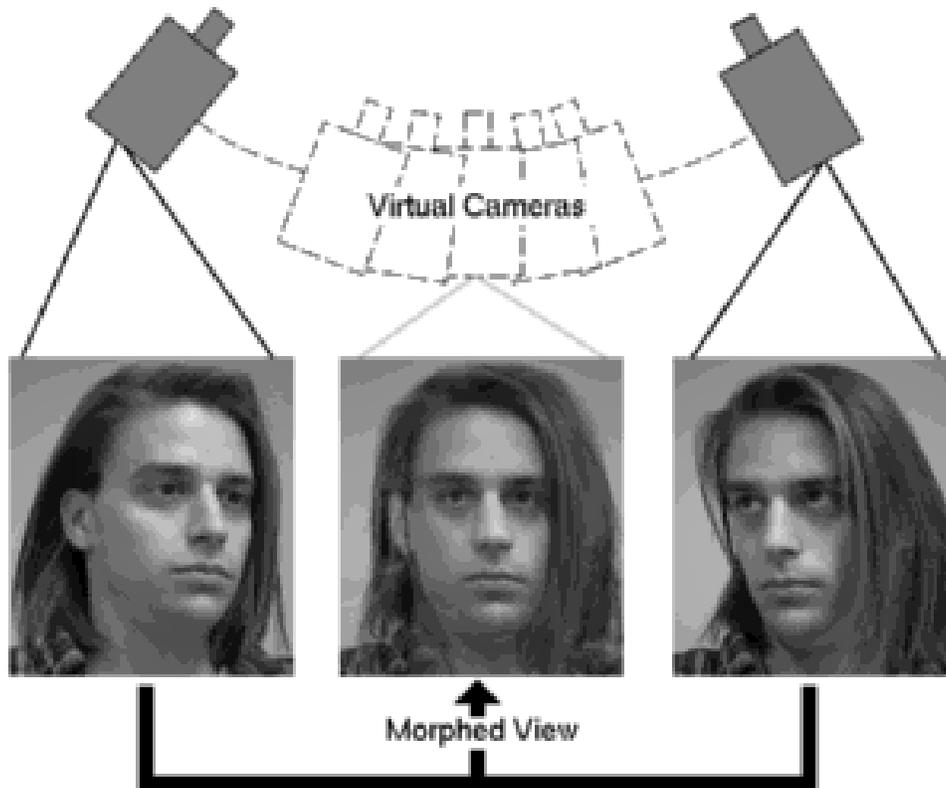
La clave para que la imagen se deforme de manera coherente con la malla de triángulos es conseguir que cuando un triángulo de la malla cambie de forma, el trozo de imagen que queda en su interior se ‘redistribuya’ sobre el mismo triángulo. Esto resulta muy fácil de implementar representando la imagen como una textura bidimensional sobre una malla plana de triángulos (el uso de texturas se describirá en el tema 7). El *morphing* se produce automáticamente cuando se mueven los vértices sin alterar las coordenadas de textura originales. Si lo que se desea es hacer más grande una parte del objeto representado en la imagen, se deberán ‘estirar’ los triángulos que lo cubren; si lo que se quiere es hacerla más pequeña habrá que disminuir el tamaño de los correspondientes triángulos. Se puede también unir vértices, con lo que ciertos triángulos (y la correspondiente parte de la imagen) desaparecerán completamente.



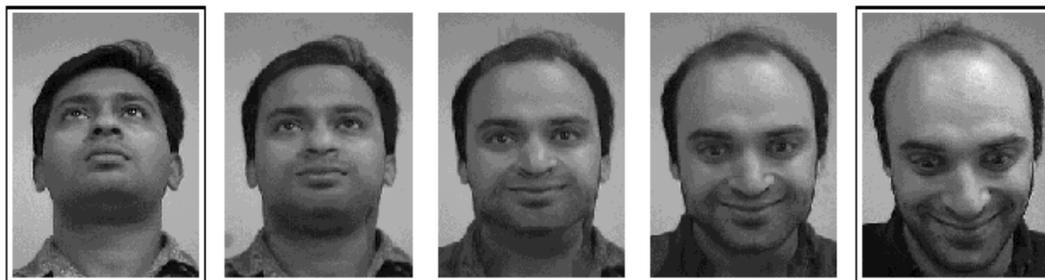
Para conseguir efectos interesantes de *morphing* cada rasgo relevante de la imagen debe quedar cubierto por triángulos de manera precisa. Por ejemplo, si queremos manipular la forma de una nariz en la imagen de una cabeza humana, tendremos que recubrir la nariz con triángulos que marcan claramente su frontera. Un caso especial de este tipo de *morphing* es el que se realiza para simular el movimiento de la cámara desde un lugar a otro, partiendo de dos imágenes, la inicial y la final, tomadas desde diferentes puntos de vista.

La forma clásica de realizar un **fundido** entre dos imágenes es interpolarlas en el tiempo, dando progresivamente más peso a una de las imágenes y menos a la otra. Pero este sistema no realiza una interpolación de las formas de los objetos que contengan las imágenes inicial y final. Podemos utilizar el *morphing* 2D para ir distorsionando las dos imágenes al mismo tiempo que fundimos la información de color. Para ello tenemos que definir mallas triangulares en ambas imágenes y establecer una

correspondencia entre los vértices. Lo más sencillo es buscar una correspondencia estructural, de manera que el vértice correspondiente a un rasgo (por ejemplo, la punta de la nariz) se corresponda en ambas imágenes, haciendo que ambas mallas sean isomorfas o coherentes. También es posible duplicar o colapsar vértices durante el proceso.



Funcionamiento del 'view morphing'



Morphing 2D con deformación y fundido simultáneo

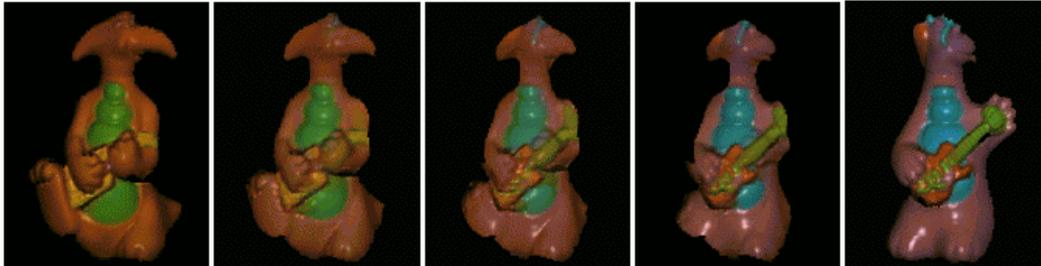
### 5.5.2 MORPHING 3D

En este caso no se trata de modificar una imagen sino la forma del objeto en tres dimensiones. Este cambio continuo puede utilizarse en animación para representar deformaciones o crear efectos visuales. Otro uso, más técnico, consiste en suavizar las transiciones entre diferentes representaciones de un mismo objeto cuando éstas tienen diferente nivel de detalle (las técnicas de nivel de detalle las presentaremos en el tema 6).

Aunque existen técnicas de morphing para otros tipos de representaciones, la manera más común de hacerlo es definiendo los objetos, al igual que en 2D, mediante una malla de triángulos. La interpolación

de la posición entre la situación inicial y final se realizará en este caso en el espacio tridimensional. También pueden interpolarse otras propiedades del objeto, como la textura o el material.

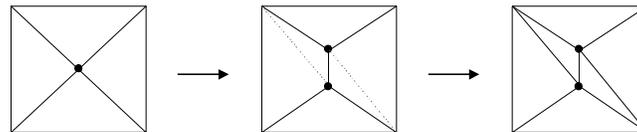
Igual que en el caso bidimensional, las dos estructuras de vértices pueden ser isomorfas, o podemos añadir o eliminar vértices durante el proceso. Esta última opción se usa cuando queremos efectuar la transición entre dos niveles de detalle diferentes del mismo objeto, que están caracterizados precisamente por tener un número diferente de vértices.



**Synthesized images between the two objects**

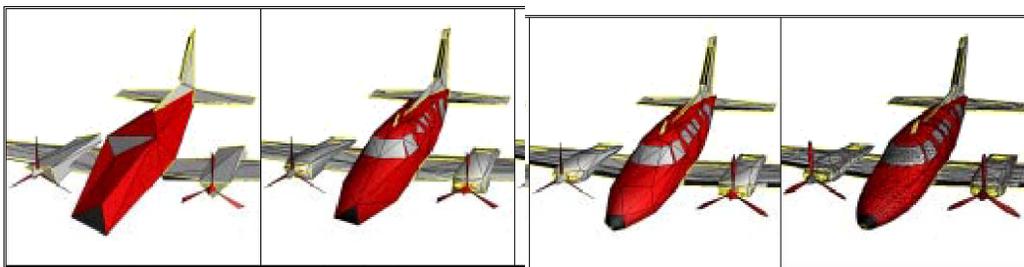
Ejemplo de cambio de la forma tridimensional entre dos muñecos

Existen diferentes formas de añadir o quitar vértices de forma continua. Una manera de hacerlo es acercar (o separar) progresivamente los dos vértices. Debemos tener en cuenta que al crear nuevos vértices debemos reconstruir la triangulación de la superficie.



Adición progresiva de un nuevo vértice para aumentar el nivel de detalle, con retriangulación

Recientemente esta técnica se ha extendido para construir la llamada **representación progresiva** de objetos poligonales, que permite el control continuo de nivel de detalle. La representación progresiva de un objeto consiste en una malla triangular básica, la de menor detalle, más una lista de vértices que deben irse añadiendo para pasar a versiones más detalladas del objeto (más la información sobre su posición final de esos vértices).



(a) Base mesh  $M^0$  (150 faces) (b) Mesh  $M^{1/5}$  (500 faces) (c) Mesh  $M^{2/5}$  (1,000 faces) (d) Original  $M=M^n$  (13,546 faces)  
 Figure 5: The PM representation of an arbitrary mesh  $M$  captures a continuous-resolution family of approximating meshes  $M^0 \dots M^n = M$ .

Ejemplo de representación progresiva de un objeto, con la posibilidad de hacer morphing entre niveles

### 5.5.3 CAMBIOS DE FORMA INDEPENDIENTES DE LA REPRESENTACIÓN

Las técnicas de morphing descritas tienen dos problemas. El primero es que resulta costoso generar de forma sencilla cambios globales en la forma del objeto (es más útil para especificar cambios en los

detalles). La segunda desventaja es que la descripción del un cambio de forma solo es aplicable a un cierto objeto, y en teniendo en cuenta determinada representación, por ejemplo la poligonal.

Vamos a ver ahora otras técnicas para describir cambios de forma, que pueden aplicarse independientemente del sistema de representación y no están ligadas a la forma concreta del objeto. Para especificar y visualizar estas deformaciones se suele utilizar un objeto geométrico sencillo (p. ej. un cubo) que envuelve al objeto real. El objeto envolvente se deforma, y el objeto real sufre la misma transformación según su posición dentro de la figura envolvente.

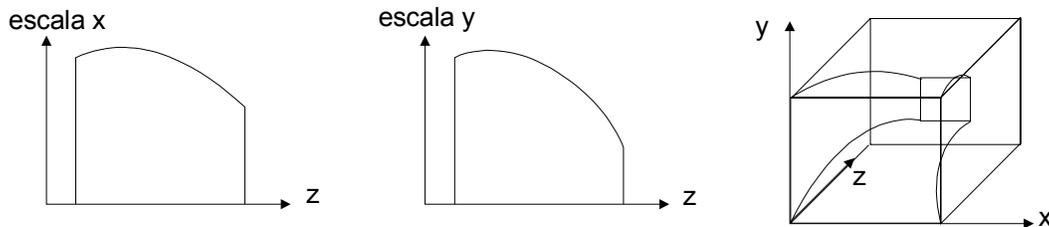
- **Modelo de deformaciones no lineales globales ( Barr )**

Hemos visto que mediante la aplicación de matrices a las coordenadas espaciales de los vértices de un objeto es posible generar transformaciones del tamaño y la posición del objeto llamadas transformaciones lineales. Las tres transformaciones lineales más utilizadas son el escalado, la traslación y la rotación. Sin embargo, estas transformaciones lineales no alteran la forma básica del objeto y cada arista recta del objeto continúa siendo recta después de la transformación.

La idea del método de Barr es utilizar estas mismas transformaciones elementales, pero aplicándolas de manera diferente a cada punto del objeto, dependiendo de sus coordenadas. Como resultado, el efecto de la operación sobre la forma global del objeto es una transformación no-lineal. Es posible escoger diferentes conjuntos de transformaciones no lineales tomando unas u otras transformaciones básicas y haciéndolas depender de las coordenadas x, y, z de diversas maneras. Algunos ejemplos podrían ser:

**Comprimir** : se utiliza el escalado de dos de las coordenadas en función de la tercera; por ejemplo, el escalado para cada punto (x, y) variando en función de z:

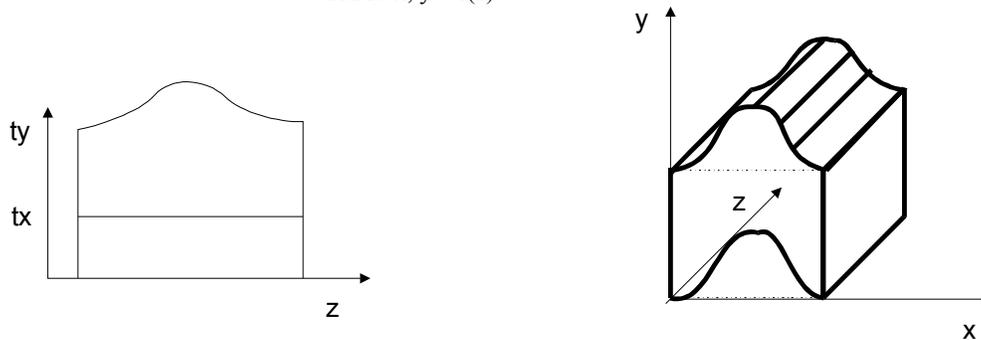
$$\begin{aligned} \text{ESCALA } x &= f(z) \\ \text{ESCALA } y &= f(z) \end{aligned}$$



**Retorcer** : se aplica una rotación a dos de las coordenadas que depende de un ángulo que a su vez es función de la coordenada restante.

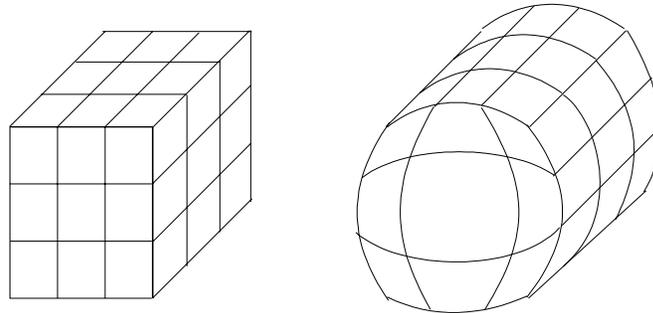
**Doblar** : en este caso es la traslación de dos de las coordenadas la que varía en función de una de la otra.

$$\text{TRAS } x, y = f(z)$$



- **Deformaciones con volúmenes paramétricos (Free-Form Deformation ó FFD)**

Es una deformación basada en la transformación de un volumen definido de forma paramétrica, como por ejemplo el hipertrozo tricúbico de Bèzier, en el que cada lado es un trozo cúbico de Bèzier definido por dos curvas cúbicas. O sea, cada arista viene definida por cuatro puntos de control: los dos vértices del cubo más dos puntos internos en la arista. En la posición de reposo los puntos de control se encuentran en las caras de un cubo.



Cada vértice en el interior del cubo tiene unas coordenadas paramétricas inducidas por la posición de los puntos de control. Por tanto, para cada posición de los puntos de control existe una relación:

$$(x, y, z) = f(u, v, w) \Rightarrow (u, v, w) = f^{-1}(x, y, z)$$

El proceso de deformación funciona de la siguiente forma:

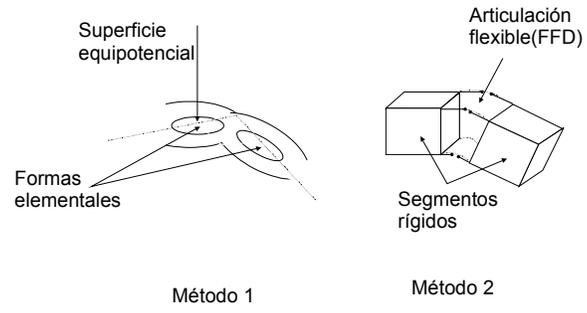
- a) En la posición de reposo se calculan las coordenadas paramétricas de cada vértice ( $u_0, v_0, w_0$ ).
- b) Cada vez que se muevan los puntos de control del hipertrozo, la relación entre coordenadas paramétricas y cartesianas viene dada por una nueva función  $f'$ . Lo que hacemos es suponer que las coordenadas paramétricas de cada punto permanecen invariantes y recalculamos las coordenadas cartesianas, lo que producirá un cambio en la forma del objeto:

$$(x', y', z') = f'(u_0, v_0, w_0)$$

#### 5.5.4 ARTICULACIONES FLEXIBLES ( *SKINNING* )

La representación visual de segmentos rígidos en una figura articulada plantea el inconveniente de que la juntura existente entre cada pareja de segmentos, no resulta natural, sobre todo cuando intentamos representar cuerpos flexibles.

Para solucionarlo tenemos dos técnicas. La primera se basa en un modelo de músculos que actúan como formas elementales para generar una superficie equipotencial que represente el papel de 'piel', creando una superficie continua que da una mejor apariencia a la articulación. La segunda técnica emplea volúmenes paramétricos concatenados (hipertrozos como los que hemos visto en el método FFD). Uno de estos hipertrozos estará conectado a los dos segmentos rígidos (ver figura 5.5.4.) y resultará deformado por el movimiento de la articulación. Mediante el método FFD calcularemos la correspondiente transformación suave de una superficie intermedia que representa el tramo de la articulación.



Métodos básicos para la definición de articulaciones flexibles de apariencia continua.