

SIMULACIÓN Y ANÁLISIS DE LA DINÁMICA DE UNA BICICLETA



Por Manuel Cornax Atienza

Tutor : Doctor Jose Luis Escalona

INDICE:

1.- INTRODUCCIÓN

2.- BREVE DESCRIPCIÓN DEL MODELO

3.- OBJETIVOS A ALCANZAR EN LA SIMULACIÓN

4.- IMPLEMENTACIÓN DE LA ESCENA EN 3D

5.- IMPLEMENTACIÓN DEL MODELO

5.1.- IMPLEMENTACIÓN EN SIMULINK

5.2.- IMPLEMENTACIÓN EN MATLAB

5.3.- BREVE DESCRIPCIÓN DE LOS ARCHIVOS NECESARIOS PARA LA ANIMACIÓN.

5.4.- ¿ INTERFAZ SIMULINK Ó INTERFAZ MATLAB?

6.- ALGUNAS NOCIONES BÁSICAS SOBRE EL COMPORTAMIENTO DE LAS BICICLETAS.

7.- CONTROL DE LA BICICLETA

7.1.- CONTROL IMPONIENDO EL ÁNGULO DEL MANILLAR

7.2.- CONTROL MEDIANTE UN MOMENTO EN EL EJE Z DEL MANILLAR

7.3.- CONTROL MEDIANTE LA INCLINACIÓN DEL CUERPO DEL CICLISTA.

7.3.1.- CÁLCULO DE LOS PARÁMETROS DEL CUERPO DEL CICLISTA

7.3.2.-ACERCA DEL PROGRAMA

7.3.3.- INTERPRETACIÓN DE LOS RESULTADOS

7.4.-CONTROL COMBINADO: MOMENTO EN EL MANILLAR Y GIRO DEL CUERPO

8.- ANÁLISIS MEDIANTE LA SIMULACIÓN DEL COMPORTAMIENTO DE LA BICICLETA

8.1.-BICICLETA A BAJAS VELOCIDADES

8.2.- DISTINTOS ÁNGULOS DE AVANCE PARA LA HORQUILLA DE

LA BICICLETA

8.2.1.- Ángulo de avance menor que el inicial

8.2.2.- Ángulo de avance mayor que el inicial

9.- CONCLUSIONES

10.- MEJORAS FUTURAS

11.- DESCRIPCIÓN DE LA DOCUMENTACIÓN ADJUNTA

12.- BIBLIOGRAFÍA

13.- REFERENCIAS

1.- INTRODUCCIÓN.

Hoy en día, en un mundo cada vez más tecnológico, resulta habitual no comprender el funcionamiento de los mecanismos y objetos que nos rodean y que utilizamos con frecuencia. Es fácil que el usuario desista en el caso de complejos ingenios como pueden ser un ordenador, un avión o un coche, todos fruto del esfuerzo de técnicos especializados en materias ajenas a él.

Sin embargo, existen otro tipo de ingenios que han evolucionado lentamente, mediante la introducción de mejoras basadas en la experiencia, y que a dicho usuario no le parecen tan ajenos a su entendimiento. Tal es el caso de la bicicleta. Todo el mundo cree conocer su funcionamiento, pero cuando son preguntados, por ejemplo, acerca de qué orden de acciones ejecutan para poder realizar un giro, muy pocos son los que responden correctamente.

Lo anterior no hace sino más atractivo su estudio, ya que aunque el margen de mejora en el diseño de una bicicleta es más escaso que en otros aparatos, es curioso saber la complejidad del comportamiento de un vehículo que mucha gente domina desde la infancia, y conocer el comportamiento de la bicicleta como sistema físico es conocer parte del entorno, ya que esta es un objeto de lo más cotidiano.

El presente proyecto tiene como objetivo el conseguir la simulación dinámica interactiva de una bicicleta de la forma más realista posible, partiendo del modelo de Whipple^[1]. Para ello se controlará de la forma más intuitiva posible, intentando no introducir artificios ni controladores que se alejen de las acciones que un humano normalmente realiza en una bicicleta.

Con dicha simulación lo que se pretende es validar el modelo matemático de la bicicleta así como comprobar el grado de importancia de los parámetros físicos que intervienen.

La bicicleta resulta un sistema mecánico interesante debido a su comportamiento no intuitivo y a que es un vehículo aparentemente sencillo y omnipresente en nuestras vidas, aun así el construir un simulador es algo más que un hito en la comprensión de la misma, ya que supone un ejercicio de lo que hoy en día es una de las grandes tendencias en la ingeniería: la simulación de sistemas mecánicos como parte del diseño. Un caso de simulación muy cercano al de la bicicleta es la simulación de automóviles, que actualmente está en pleno auge. Prueba de ello es el inmenso simulador elaborado por Toyota en 2007 y constituido por una esfera de 7 metros de alto con 360° de pantalla y un sistema hidráulico que reproduce las físicas de la conducción.

Con este tipo de simulaciones se ensayan los vehículos incluso antes de ser construidos, ahorrando costes en cuanto a prototipos o probándolos en situaciones

excepcionales para evaluar la seguridad de los mismos. Por lo tanto la elaboración de simuladores escapa a la mera curiosidad y constituye un elemento eficaz en el análisis y control de sistemas mecánicos.

El proyecto parte de un programa formado por un diagrama de bloques de Simulink en el que se integran las ecuaciones de la bicicleta con un paso de tiempo variable y en el que la animación producida por el mismo es un simple 'plot' de MATLAB. En este programa inicial la entrada al sistema es un momento sobre el manillar de la bicicleta, el cual es introducido en el diagrama de bloques y cuya señal procede de un joystick. Cabe destacar que en esta situación de partida resulta imposible seguir un camino predefinido, por lo que se tendrán que ensayar distintas formas de control hasta obtener una experiencia de control aceptable.

2.- BREVE DESCRIPCIÓN DEL MODELO.

Debido a que el modelado matemático de la bicicleta no forma parte de este proyecto se procede a una muy breve descripción del mismo. El documento en el cual se basa dicho modelo figura al final del proyecto, en la referencia [2].

En este modelo inicial se han tenido en cuenta una serie de hipótesis:

- El suelo se considera horizontal, no habiendo ningún tipo de pendiente.
- El cuerpo del ciclista se ha considerado rígido y en el plano de la bicicleta.
- El par correspondiente al pedaleo no considera las oscilaciones en el momento debidas a la desigual aplicación de la fuerza en el transcurso de una pedalada.
- Las ruedas son consideradas rígidas y su contacto con el suelo como un único punto.
- Las ruedas ruedan sin deslizar.
- No existe fricción interna en los mecanismos ni otros efectos disipativos que no sean el debido al aire.

El modelo consiste en la descripción de la bicicleta como un sistema de 5 sólidos rígidos, los cuales son:

- 1.- El suelo.
- 2.- La rueda trasera.
- 3.- El cuadro.
- 4.- El manillar.
- 5.- La rueda delantera.

Las coordenadas usadas y la geometría de la bicicleta se pueden apreciar en la figura 1.

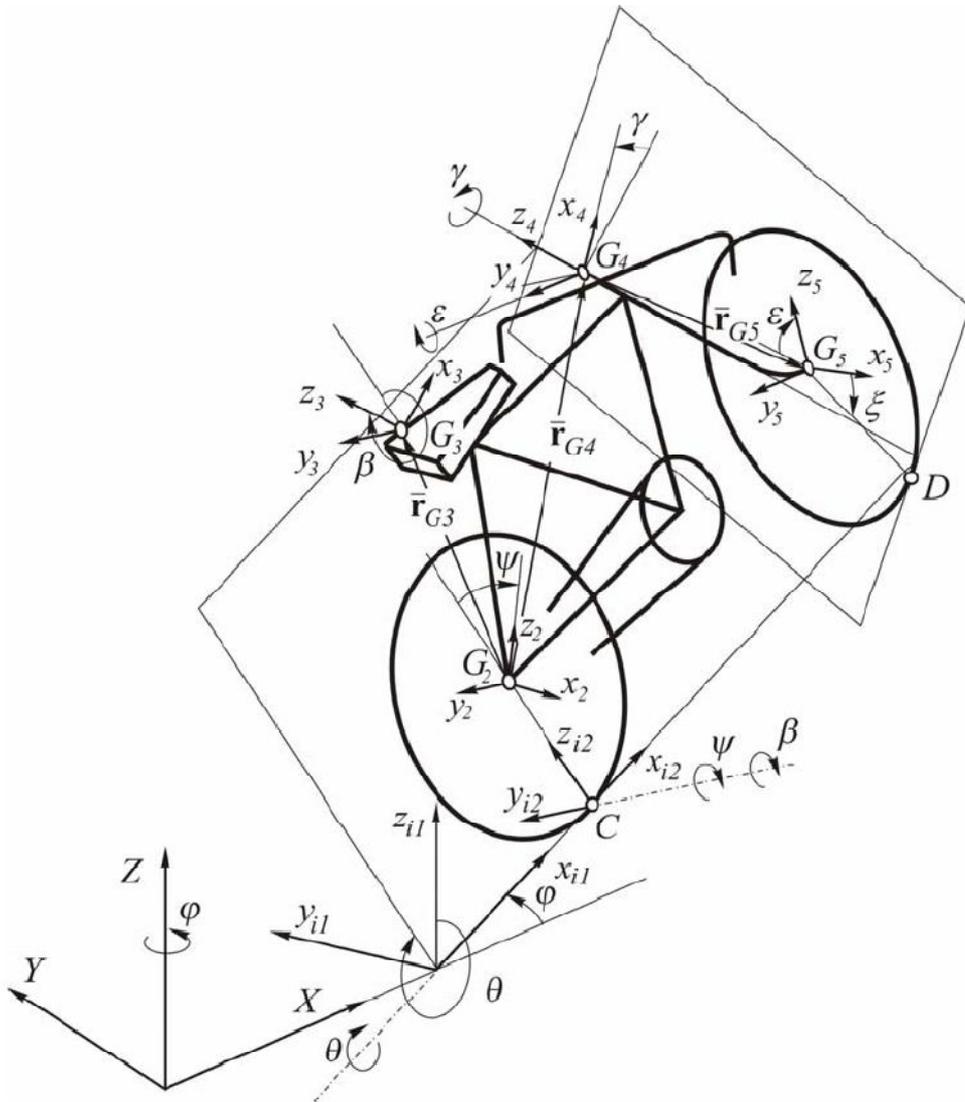


Figura 1

El sistema inicial se describe cinemáticamente mediante un conjunto de 9 coordenadas, las dos primeras de las cuales son las referentes a la posición en el plano del suelo del punto de contacto de la rueda trasera de la bicicleta, el resto de las coordenadas son giros en sistemas de referencia locales. La descripción del modelo se ha realizado de tal forma que a cada giro de los antes mencionados le corresponda un sólido o sistema de coordenadas local, de manera que las coordenadas absolutas de un punto de cualquier sólido se puedan expresar como las coordenadas locales del punto multiplicadas por las sucesivas matrices de giro de los sólidos anteriores (más la

posición en globales del punto de contacto de la rueda trasera). Para ello ha sido necesario introducir dos sólidos auxiliares a los que se ha llamado sólido intermedio 1 y sólido intermedio 2.

Las coordenadas utilizadas para describir la posición y orientación de la bicicleta son:

1. Las coordenadas x_c e y_c que marcan la posición del punto de contacto C de la rueda trasera en el plano $\langle X Y \rangle$ del sistema global de referencia.
2. El ángulo de guiñada (yaw) φ que forma el eje x_{i1} del sistema intermedio de referencia con el eje X del sistema global de referencia.
3. El ángulo de vuelco (roll) θ que forma el eje z_{i2} del segundo sistema intermedio de referencia con el eje z_{i1} del primer sistema intermedio de referencia. Los ángulos φ y θ determinan la orientación del plano que contiene al cuadro de la bicicleta.
4. El ángulo de rodadura (pitch) ψ que forma el eje z_2 del sistema de referencia asociado a la rueda trasera con el eje z_{i2} del segundo sistema intermedio de referencia.
5. El ángulo β que forma el eje z_3 del sistema de referencia del marco con el eje z_{i2} del segundo sistema intermedio de referencia.
6. El ángulo de dirección γ que forma el eje x_4 del sistema de referencia del manillar con el eje x_3 del sistema de referencia del marco.
7. El ángulo de rodadura (pitch) ε que forma el eje z_5 del sistema de referencia asociado a la rueda delantera con el eje z_4 del manillar.

Finalmente, las coordenadas descritas se pueden agrupar en el vector:

$$\mathbf{q} = [x_c \quad y_c \quad \varphi \quad \theta \quad \psi \quad \beta \quad \gamma \quad \varepsilon]^T$$

Las matrices de giro de los distintos sistemas de referencia con respecto al sistema global de coordenadas se calculan como:

$$\begin{aligned} \mathbf{A}^{i1} &= \mathbf{A}_\varphi, \quad \mathbf{A}^{i2} = \mathbf{A}_\varphi \mathbf{A}_\theta \\ \mathbf{A}^2 &= \mathbf{A}_\varphi \mathbf{A}_\theta \mathbf{A}_\psi, \quad \mathbf{A}^3 = \mathbf{A}_\varphi \mathbf{A}_\theta \mathbf{A}_\beta, \\ \mathbf{A}^4 &= \mathbf{A}_\varphi \mathbf{A}_\theta \mathbf{A}_\beta \mathbf{A}_\gamma, \quad \mathbf{A}^5 = \mathbf{A}_\varphi \mathbf{A}_\theta \mathbf{A}_\beta \mathbf{A}_\gamma \mathbf{A}_\varepsilon \end{aligned} \tag{1}$$

siendo:

$$\mathbf{A}_\varphi = \begin{bmatrix} \cos\varphi & -\text{sen}\varphi & 0 \\ \text{sen}\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{A}_\theta = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\text{sen}\theta \\ 0 & \text{sen}\theta & \cos\theta \end{bmatrix},$$

$$\mathbf{A}_\psi = \begin{bmatrix} \cos\psi & 0 & \text{sen}\psi \\ 0 & 1 & 0 \\ -\text{sen}\psi & 0 & \cos\psi \end{bmatrix}, \quad \mathbf{A}_\beta = \begin{bmatrix} \cos\beta & 0 & \text{sen}\beta \\ 0 & 1 & 0 \\ -\text{sen}\beta & 0 & \cos\beta \end{bmatrix},$$

$$\mathbf{A}_\gamma = \begin{bmatrix} \cos\gamma & -\text{sen}\gamma & 0 \\ \text{sen}\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{A}_\varepsilon = \begin{bmatrix} \cos\varepsilon & 0 & \text{sen}\varepsilon \\ 0 & 1 & 0 \\ -\text{sen}\varepsilon & 0 & \cos\varepsilon \end{bmatrix}$$

$$\mathbf{A}_\varphi = \begin{bmatrix} \cos\varphi & -\text{sen}\varphi & 0 \\ \text{sen}\varphi & \cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{A}_\theta = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\text{sen}\theta \\ 0 & \text{sen}\theta & \cos\theta \end{bmatrix},$$

$$\mathbf{A}_\psi = \begin{bmatrix} \cos\psi & 0 & \text{sen}\psi \\ 0 & 1 & 0 \\ -\text{sen}\psi & 0 & \cos\psi \end{bmatrix}, \quad \mathbf{A}_\beta = \begin{bmatrix} \cos\beta & 0 & \text{sen}\beta \\ 0 & 1 & 0 \\ -\text{sen}\beta & 0 & \cos\beta \end{bmatrix},$$

$$\mathbf{A}_\gamma = \begin{bmatrix} \cos\gamma & -\text{sen}\gamma & 0 \\ \text{sen}\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \mathbf{A}_\varepsilon = \begin{bmatrix} \cos\varepsilon & 0 & \text{sen}\varepsilon \\ 0 & 1 & 0 \\ -\text{sen}\varepsilon & 0 & \cos\varepsilon \end{bmatrix}$$

Para obtener el término correspondiente a la energía cinética en las ecuaciones de Lagrange es necesario obtener los valores de la velocidad de los centros de gravedad y la velocidad angular de cada sólido en coordenadas globales.

La velocidad angular en locales, $\bar{\omega}$, o en globales, ω , de cada sólido se podrá expresar mediante el uso de la matrices $\bar{\mathbf{G}}$ o \mathbf{G} respectivamente, en las que cada columna está formada por el valor en el sistema de coordenadas local o global de los vectores directores correspondientes a los giros que experimenta el sólido en cuestión.

$$\omega^2 = \mathbf{G}^2 \dot{\mathbf{q}} \quad (2)$$

$$\bar{\omega}^2 = \bar{\mathbf{G}}^2 \dot{\mathbf{q}} \quad (3)$$

Por ejemplo, para el sólido 2:

$$\omega^2 = \varphi \mathbf{k} + \theta \mathbf{i}^{i1} + \psi \mathbf{j}^{i2}, \quad (4)$$

$$\mathbf{k} = [\mathbf{A}^1]_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad \mathbf{i}^{i1} = [\mathbf{A}^{i1}]_1 = \begin{bmatrix} \cos\varphi \\ \text{sen}\varphi \\ 0 \end{bmatrix}, \quad \mathbf{j}^{i2} = [\mathbf{A}^{i2}]_2 = \begin{bmatrix} -\text{sen}\varphi \cos\theta \\ \cos\varphi \cos\theta \\ \text{sen}\theta \end{bmatrix}$$

$$\mathbf{G}^2(\mathbf{q}) = \begin{bmatrix} 0 & 0 & 0 & \cos\varphi & -\text{sen}\varphi\cos\theta & 0 & 0 & 0 \\ 0 & 0 & 0 & \text{sen}\varphi & \cos\varphi\cos\theta & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & \text{sen}\theta & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & [\mathbf{A}^1]_3 & [\mathbf{A}^{i1}]_1 & [\mathbf{A}^{i2}]_2 & \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

En cuanto a la velocidad de los centros de gravedad, se usará la matriz jacobiana de la posición de los centros de gravedad en globales, \mathbf{H}^i , de manera que:

$$\mathbf{v}_{G_i} = \mathbf{H}^i \dot{\mathbf{q}}, \quad \mathbf{H}^i = \frac{\partial \mathbf{r}_{G_i}}{\partial \mathbf{q}} \quad (5)$$

Siendo \mathbf{r}_{G_i} la posición en globales del centro de gravedad del sólido. La energía cinética del sistema nos queda:

$$T = \sum_{i=2}^5 \frac{1}{2} \left[m^i (\mathbf{v}_{G_i})^T \mathbf{v}_{G_i} + (\bar{\boldsymbol{\omega}}^i)^T \bar{\mathbf{I}}^i \bar{\boldsymbol{\omega}}^i \right] \quad (6)$$

$$T = \sum_{i=2}^5 \frac{1}{2} \left[m^i \dot{\mathbf{p}}^T \mathbf{H}^i \mathbf{H}^i \dot{\mathbf{p}} + \dot{\mathbf{p}}^T \bar{\mathbf{G}}^i \bar{\mathbf{I}}^i \bar{\mathbf{G}}^i \dot{\mathbf{p}} \right] = \sum_{i=2}^5 \frac{1}{2} \dot{\mathbf{p}}^T \left[m^i \mathbf{H}^i \mathbf{H}^i + \bar{\mathbf{G}}^i \bar{\mathbf{I}}^i \bar{\mathbf{G}}^i \right] \dot{\mathbf{p}}$$

El sistema cuenta también con tres tipos de restricciones: Las de contacto y las de rodadura sin deslizamiento correspondientes a las ruedas y la de movilidad. En total 7 restricciones, con lo que contamos con 2 grados de libertad.

- Las de contacto consisten en imponer que la coordenada vertical de la posición de contacto de cada rueda y de la tangente al contacto sean nulas.

- Las de rodadura sin deslizamiento implica que la velocidad del contacto de las ruedas con el suelo es en todo momento nula. O lo que es lo mismo, que:

$$\mathbf{C}^{rod}(\mathbf{p}, \dot{\mathbf{p}}) = \begin{bmatrix} \mathbf{C}^{rod,2}(\mathbf{p}, \dot{\mathbf{p}}) \\ \mathbf{C}^{rod,5}(\mathbf{p}, \dot{\mathbf{p}}) \end{bmatrix} = \mathbf{0}, \quad \mathbf{B}(\mathbf{p}) = \begin{bmatrix} \mathbf{B}^2(\mathbf{p}) \\ \mathbf{B}^5(\mathbf{p}) \end{bmatrix} \quad (7)$$

Donde las matrices \mathbf{C} vienen dadas por las ecuaciones siguientes.

$$\left. \begin{array}{l} \dot{x}_c - R \cos \varphi \dot{\psi} = 0 \\ \dot{y}_c - R \text{sen} \varphi \dot{\psi} = 0 \end{array} \right\} \Rightarrow \mathbf{C}^{rod,2}(\mathbf{p}, \dot{\mathbf{p}}) = \mathbf{0} \quad (8)$$

$$\left. \begin{array}{l} [\mathbf{v}_D]_X = 0 \\ [\mathbf{v}_D]_Y = 0 \end{array} \right\} \Rightarrow \mathbf{C}^{rod,5}(\mathbf{p}, \dot{\mathbf{p}}) = \mathbf{0} \quad (9)$$

Y las matrices \mathbf{B} son el Jacobiano de cada matriz \mathbf{C} correspondiente con respecto al vector de coordenadas.

- La restricción de movilidad es la siguiente:

$$\psi - \frac{V}{R}t = 0 \Rightarrow \mathbf{C}^{mov}(\mathbf{q}, t) = 0 \quad (10)$$

Todas las ecuaciones de restricción se agrupan en $\mathbf{C}(\mathbf{p}, \dot{\mathbf{p}}, t)$ como sigue:

$$\mathbf{C}(\mathbf{p}, \dot{\mathbf{p}}, t) = \begin{bmatrix} \mathbf{C}^{con}(\mathbf{p}) \\ \mathbf{C}^{rod}(\mathbf{p}, \dot{\mathbf{p}}) \\ \mathbf{C}^{mov}(\mathbf{p}, t) \end{bmatrix} = \mathbf{0}, \quad (11)$$

Haciendo uso de las ecuaciones de Lagrange nos queda un sistema de ecuaciones diferenciales algebraicas al cual se le aplica el método de *generalized coordinate partitioning* para transformar las ecuaciones en un sistema ODE. Este método consiste en las siguientes operaciones:

$$\begin{bmatrix} \mathbf{C}_p^{con} \\ \mathbf{B} \\ \mathbf{C}_p^{mov} \end{bmatrix} \mathbf{p} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ V/R \end{bmatrix} \Rightarrow \mathbf{D}\mathbf{p} = -\mathbf{C}_t \quad (12)$$

En la ecuación de restricción anterior se despejan las coordenadas dependientes poniéndolas en función de las independientes.

$$\dot{\mathbf{p}}_d = \mathbf{D}_d^{-1}(-\mathbf{D}_i\dot{\mathbf{p}}_i - \mathbf{C}_t) \quad (13)$$

$$\dot{\mathbf{p}} = \begin{bmatrix} \dot{\mathbf{p}}_i \\ \dot{\mathbf{p}}_d \end{bmatrix} = \begin{bmatrix} \mathbf{I} \\ -\mathbf{D}_d^{-1}\mathbf{D}_i \end{bmatrix} \dot{\mathbf{p}}_i + \begin{bmatrix} \mathbf{0} \\ -\mathbf{D}_d^{-1}\mathbf{C}_t \end{bmatrix} = \mathbf{E}\dot{\mathbf{p}}_i + \mathbf{F} \quad (14)$$

Se calcula la derivada segunda del vector de coordenadas.

$$\ddot{\mathbf{p}} = \mathbf{E}\ddot{\mathbf{p}}_i + \dot{\mathbf{E}}\dot{\mathbf{p}}_i + \dot{\mathbf{F}} = \mathbf{E}\ddot{\mathbf{p}}_i + \mathbf{J} \quad (15)$$

Finalmente se sustituyen los vectores $\dot{\mathbf{p}}$ y $\ddot{\mathbf{p}}$ de las ecuaciones 13 y 14 en las ecuaciones del sistema y tras despejar nos queda:

$$\mathbf{E}^T \mathbf{M} \mathbf{E} \ddot{\mathbf{p}}_i = \mathbf{E}^T (\mathbf{Q}_v + \mathbf{Q}_{grav} + \mathbf{Q}_{ext} - \mathbf{M}\mathbf{J}) \quad (16)$$

Una vez calculadas las coordenadas dependientes este sistema ya no es de 16 ecuaciones como el inicial (9 ecuaciones debidas a las coordenadas y 7 debidas a las

restricciones), sino un sistema de tantas ecuaciones diferenciales ordinarias como coordenadas independientes de la forma $M_i \ddot{q}_i = Q_i$ (17).

Por el contrario las matrices de masa y el vector de fuerzas ahora dependen de las coordenadas dependientes, por lo que la solución numérica de las ecuaciones requiere la solución en cada paso de tiempo de las ecuaciones de restricción y sus derivadas para el cálculo de las coordenadas y velocidades dependientes a partir de las independientes y poder así obtener las matrices M_i y Q_i . La complejidad de las matrices que intervienen en el cálculo y que dependen del vector 'q' es tal que para hallarlas es preciso el uso de cálculo simbólico mediante MATLAB.

En el programa de cálculo simbólico que calcula las matrices y vectores que aparecen en las ecuaciones del sistema simplemente se definen dichas matrices y vectores como variables simbólicas y se calculan escribiendo en el programa las ecuaciones que las definen, que no son otras que las del presente capítulo.

Una vez definidas las ecuaciones del sistema el problema se centra en la implementación de una simulación adecuada a partir de estas, la cual se desarrollará en los siguientes capítulos.

En el modelo del sistema se ha incluido la fricción del aire para darle más realismo.

3.- OBJETIVOS A ALCANZAR EN LA SIMULACIÓN.

Si se quiere controlar la bicicleta de una forma intuitiva habrá que tener en cuenta los siguientes factores:

1º.- El conductor tiene que tener una percepción de la situación del sistema adecuada. Esto se puede conseguir mediante el uso de una vista subjetiva y consiguiendo una sensación adecuada de velocidad mediante la introducción de suficientes elementos que sirvan como referencia en la simulación. Asimismo el coste computacional de realizar un plotado en MATLAB para cada intervalo de tiempo parece demasiado elevado. Esto sugiere el uso de algún tipo de motor gráfico.

2º.- Para que el control sea efectivo es necesario que el sistema de ecuaciones se integre en intervalos de tiempo reales, es decir, que sean los mismos que él usuario percibe y de esta manera la sensación que da el simulador sea realista.

3º.- Debido a que la bicicleta resulta inicialmente difícil de controlar y a que en la realidad existen diversas formas de conducir una bicicleta conviene probar distintos mecanismos de control de la misma.

4.- IMPLEMENTACIÓN DE LA ESCENA EN 3D.

Para la realización de la escena en 3D se ha optado por realizar la interfaz en VRML, debido a que constituye una plataforma para la creación de escenas tridimensionales interactivas compatible con MATLAB y fácil de utilizar. Además Simulink viene con un editor de VRML preinstalado de fácil manejo.

Pese a su versatilidad, en el código en formato VRML resulta difícil definir formas complejas ya que los prototipos se definen a partir de aristas y vértices. Afortunadamente algunos programas de CAD como SolidWorks permiten exportar las figuras realizadas a VRML. De esta manera, la mayor parte de los sólidos que aparecen en la escena han sido diseñados con SolidWorks, que además nos permite un mayor control de las dimensiones de la figura mediante el uso de cotas inteligentes. Este hecho es de gran importancia, ya que en un primer momento la animación se hizo mediante VRML un conjunto de sólidos elementales introduciendo las dimensiones de forma aproximada y se comprobó que para una figura en la que la distancia entre ruedas era de 2m, en vez de 1.0323m como la bicicleta del modelo, la conducción de la misma era muy distinta debido a que la percepción que se obtiene de los giros de cada cuerpo vistos desde la posición del conductor estaba alterada. En las figuras 2, 3 y 4 se pueden ver imágenes de algunos de los modelos en SolidWorks que se han empleado tanto para la bicicleta como para el circuito.

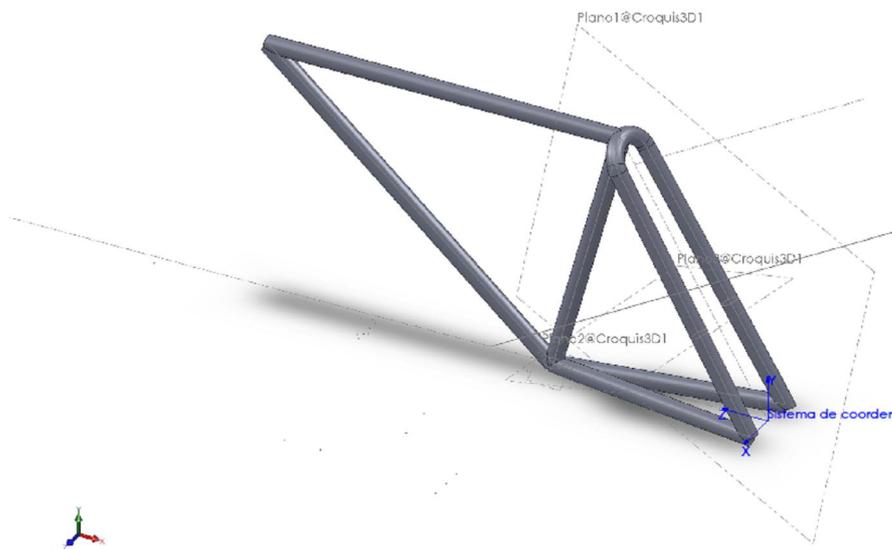


Figura 2: Cuadro de la bicicleta

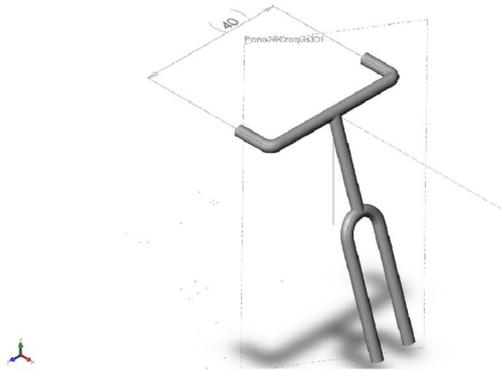


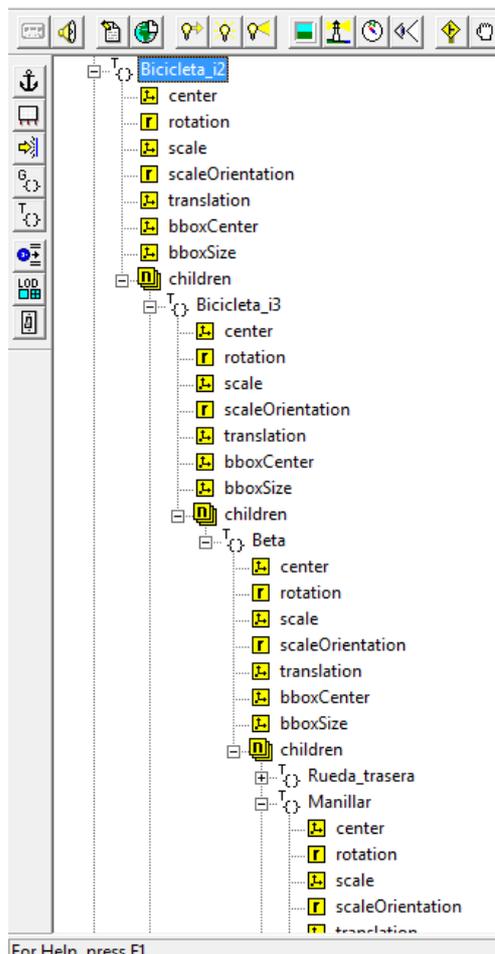
Figura 3.: Manillar y horquilla



Figura 4.: Circuito

Para realizar la escena en VRML es preciso definir una serie de nodos. Un nodo es una entidad en VRML que debe ser declarada al principio del programa y cuyas propiedades pueden ser modificadas. Entre dichas propiedades están las geométricas de cada nodo, que pueden ser especificadas mediante el uso de MATLAB o Simulink, con lo que se obtendrá la animación buscada. Estos nodos se pueden definir de forma jerárquica, de manera que los cambios que se produzcan en una posición mayor del árbol jerárquico afectaran a los nodos inferiores. Aprovechando esta cualidad se definirán los nodos de forma que cada uno equivalga a un sólido del modelo matemático. Al tener estos sólidos asociados cada uno un giro dichos giros resultarán

más fáciles de definir. A continuación se muestra el árbol jerárquico correspondiente a la bicicleta y que coordenadas corresponden los cambios que se realizarán en cada nodo:



- 1.- Mundo.
- 2.- Bicicleta_i2. Traslación x_c, y_c . Rotación φ .
- 3.- Bicicleta_i3. Rotación θ .
- 4.- Beta. Rotación β .
- 5.- Rueda_trasera. Rotación ψ .
- 6.- Manillar. Rotación γ .
- 7.- Rueda_delantera. Rotación ξ .

Figura 5.: Árbol jerárquico del editor de VRML

Este árbol jerárquico no es otro que el que surge al establecerse dependencias entre los nodos, los cuales están englobados unos dentro de otros. Por ejemplo, el nodo Bicicleta_i2 engloba tanto al nodo manillar como al Rueda trasera. Cualquier cambio que realice en Bicicleta_i2 afectará al manillar y a la Rueda trasera, no siendo esta relación recíproca.

Las traslaciones de los nodos se definen con un vector de tres coordenadas.

Las rotaciones de los nodos se definen con un vector de cuatro coordenadas en el que las tres primeras corresponden al eje del giro y la cuarta al ángulo girado. Como cada giro tiene como eje uno de los ejes coordenados del sistema de coordenadas de su sólido correspondiente, las tres primeras coordenadas de la cuaterna siempre serán un vector unitario.

El ángulo β , definido en el sentido positivo en el eje y_{i_2} , permanece prácticamente constante y con un valor, al que se llamará β_0 , que se corresponde con el ángulo de avance de la horquilla de la bicicleta definido en el sentido que se muestra en la figura 6.

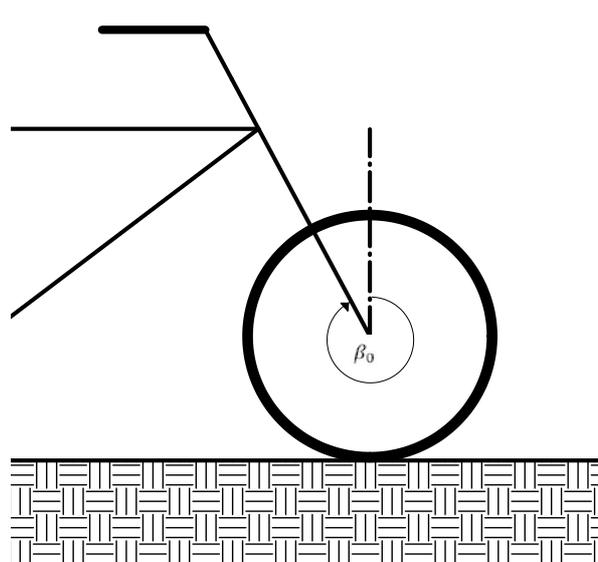


Figura 6

Para simplificar la animación se han dibujado las figuras ya giradas este ángulo β_0 , para así ahorrar el coste computacional de tener que actualizar el valor del nodo Beta cada instante.

Nótese que pese a que los nodos creados tienen correspondencia con los sólidos definidos en el modelo, no poseen los mismos nombres. Por ejemplo, el nodo Bicicleta_i3 se corresponde con el sólido i2.

El código VRML también permite incluir texto en la animación. El renderizar el texto aumenta de forma crítica el tiempo necesario para realizar la animación, por lo que dicho texto se ha limitado a un indicador de la velocidad longitudinal de la bicicleta.

Las texturas [3] que se utilizan en la animación figuran en la carpeta textures, en la misma dirección que el archivo VRML. En la figura 7 se puede ver una imagen de la escena de VRML acabada.

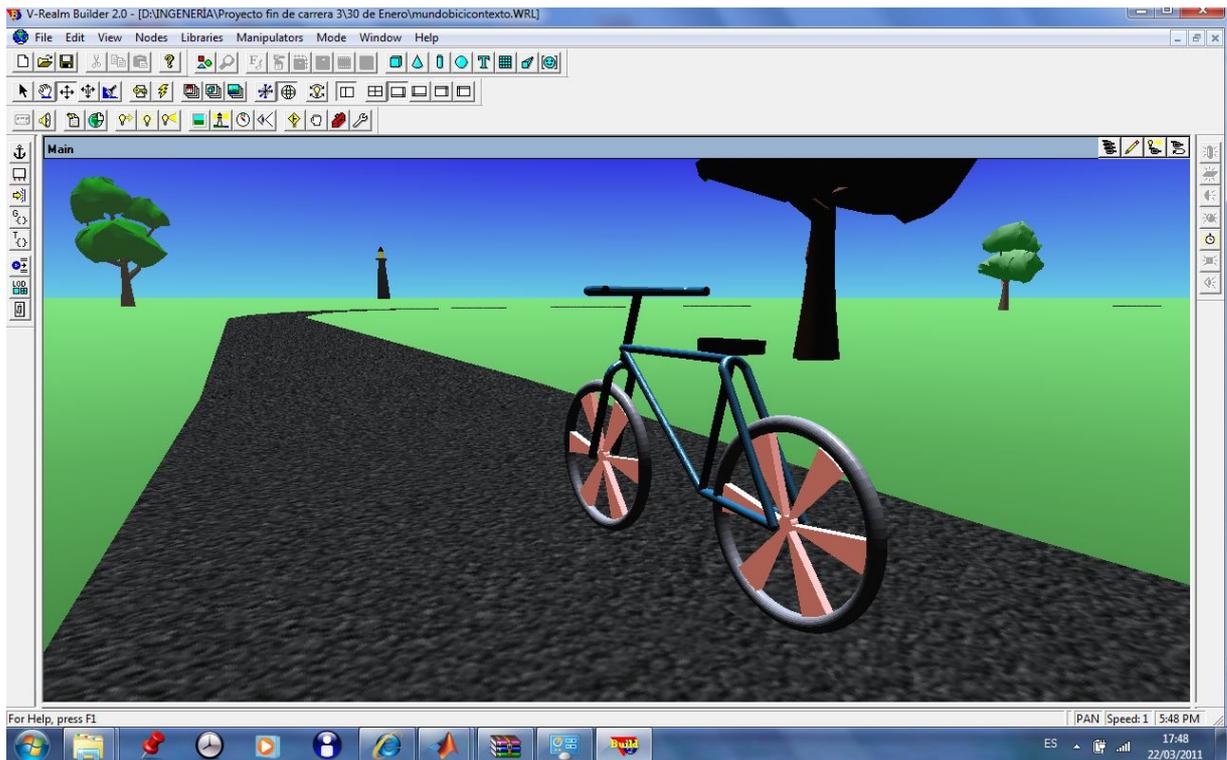


Figura 7

5.- IMPLEMENTACIÓN DEL MODELO.

Como se comenta en el apartado anterior, la escena creada en VRML puede ser modificada con los valores procedentes de la integración del modelo matemático obteniéndose así la animación deseada. Esto se puede conseguir de varias maneras: mediante un diagrama de bloques de Simulink o llamando a la escena mediante ciertos comandos en un archivo de MATLAB. También la entrada al sistema puede ser obtenida gracias al uso de la toolbox Simulink 3D, que incluye un bloque para el diagrama de Simulink que crea un objeto correspondiente al joystick, o creando el mismo objeto mediante el uso de la interfaz de Simulink 3D en MATLAB.

5.1.- IMPLEMENTACIÓN EN SIMULINK.

El diagrama de bloques de la simulación realizada en Simulink es el de la figura 8. En él se ha introducido un objeto correspondiente al joystick y la escena en VRML mediante diagramas de la toolbox Simulink 3D. Después de adecuar la señal procedente del joystick mediante una función de transferencia de primer orden y aproximar sus puntos con el bloque de Simulink Lookup table, esta se da como entrada a la función de estado Bicisfun.m. En Bicisfun se definen 10 estados continuos, que constituyen un vector formado por las 3 variables independientes (γ, θ, ψ) , 4 dependientes $(x_c, y_c, \varphi, \beta)$ y ε y las 3 derivadas de las variables independientes. A este vector se le llamará a partir de ahora \mathbf{q} .

Asimismo en dicha función de estado (Bicisfun) se definen cuales se van a ser las derivadas de los diez estados continuos. Dichas derivadas vendrán dadas por la función Derivadasbici.m, en las que se usan las ecuaciones del modelo matemático de partida de este proyecto para calcular $\dot{\mathbf{q}}$. Es decir, Derivadasbici.m hace el siguiente proceso:

$$\mathbf{q} \rightarrow \mathbf{M}_i(\mathbf{q})\mathbf{Q}_i(\mathbf{q}) \rightarrow \ddot{\mathbf{q}}_{din} = \mathbf{M}_i^{-1}\mathbf{Q}_i \quad (18)$$

En el que $\mathbf{M}_i(\mathbf{q})$ y $\mathbf{Q}_i(\mathbf{q})$ son las matrices resultantes de transformar en ODEs las ecuaciones de Lagrange y \mathbf{q} es el vector de 10 coordenadas que se comentó anteriormente.

Con el vector \mathbf{q} y su derivada definidos en la función de estado (S-function) Simulink es capaz de actualizar los estados de la misma dando el \mathbf{q} siguiente. El proceso anterior se puede entender como el que se muestra en el diagrama de la figura 9.

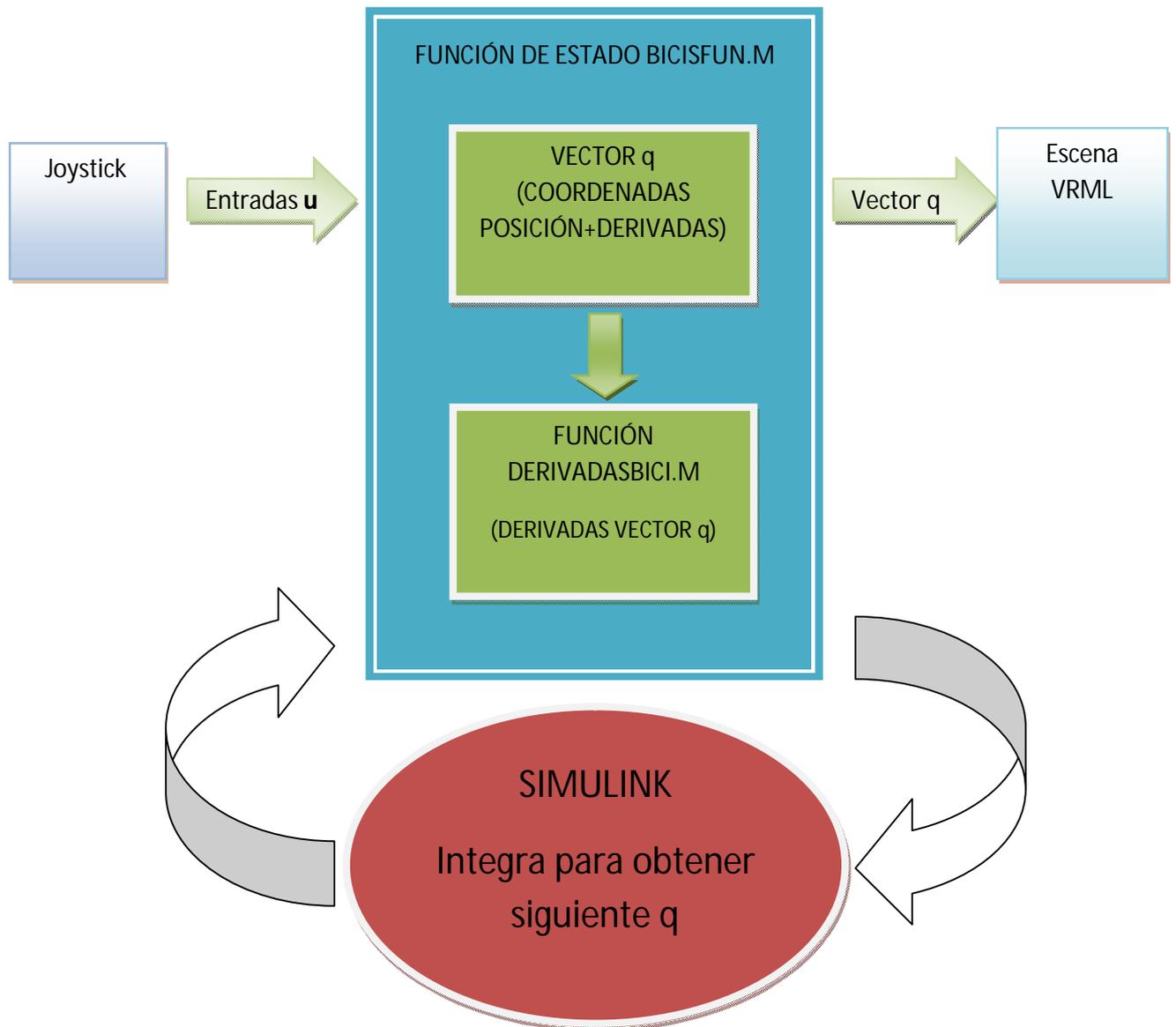


Figura 9

5.2.- IMPLEMENTACIÓN EN MATLAB.

Una forma alternativa de conseguir la animación es creando el objeto de MATLAB 'joystick' y llamando a la escena de VRML a partir de la interfaz de MATLAB.

En vez de usando una S-function, \mathbf{q} se calculará mediante un integrador explícito, en este caso un Runge-Kutta. La llamada a la función del integrador explícito se realiza en el programa 'Integrabici.m', que será el programa principal a partir del cual se llaman a las demás funciones y el que se tendrá que ejecutar para poner en marcha la

simulación. En dicho programa, por medio de los comandos tic y toc de MATLAB se obtendrá el tiempo del CPU y mediante un bucle while la integración se realizará para los sucesivos intervalos de tiempo dados por el comando toc, que se encuentra en dicho bucle y después del integrador.

La función del integrador explícito se llamará 'RungeKutta4.m' y tendrá como inputs la función a integrar, que es la definida por DerivadasBici.m, y el intervalo de tiempo dado por el comando toc.

5.3.- BREVE DESCRIPCIÓN DE LOS ARCHIVOS NECESARIOS PARA LA ANIMACIÓN.

La función Derivadasbici.m, la misma para el programa en sus versiones en MATLAB y Simulink, realiza una serie de llamadas a otras funciones que a continuación serán brevemente descritas con el fin de que se conozca la función que desempeñan en el programa.

- **rHolon.m:** Llama a las funciones encargadas de calcular las restricciones holónomas y sus derivadas. Es llamada desde Derivadasbici.m.
- **CalculaRes.m:** Calcula el valor de $\mathbf{C} = [\mathbf{r}_{az} \ \mathbf{t}_{az}]$, es decir, el valor de la componente en el eje z de los vectores posición y tangente referentes al punto de contacto de la rueda delantera con el suelo. Si \mathbf{C} no es nulo, se iterará en DerivadasBici.m hasta que se encuentre dicho punto de contacto. Esta función es llamada desde rHolon.m y desde Restr.m.
- **CalculaCq.m:** Calcula el jacobiano de las restricciones holónomas. Llamada desde rHolon.m.
- **CalculadCqdp.m:** Calcula la derivada del jacobiano de las restricciones holónomas. Llamada desde rHolon.m.
- **Restr.m:** Llama a CalculaRes.m para calcular \mathbf{C} en la iteración correspondiente a la búsqueda del punto de contacto de la rueda delantera en DerivadasBici.m.
- **rNoHolon.m:** Llama a las funciones encargadas de calcular las restricciones no holónomas y sus derivadas correspondientes. Es llamada desde Derivadasbici.m.
- **CalculaB.m:** Calcula la matriz jacobiana \mathbf{B} correspondiente a las restricciones de contacto. Es llamada desde rNoHolon.m.
- **CalculadBdp.m:** Calcula dBdp. Es llamada desde rNoHolon.m.
- **MasaFuerzas.m:** Llama a las funciones encargadas de calcular la matriz de masas del sistema y los vectores de fuerzas. Llamada desde DerivadasBici.m.
- **CalculaM.m:** Calcula la matriz de masas \mathbf{M} . Llamada desde MasaFuerzas.m.

- **CalculaQaero:** Calcula el vector de fuerzas debido al rozamiento con el aire. Llamada desde MasaFuerzas.m
- **Calcula Qgrav:** Calcula el vector de gravedad. Llamada desde MasaFuerzas.m.
- **Calcula Qv:** Calcula el vector **Qv**. Llamada desde MasaFuerzas.m.

5.4.- ¿ INTERFAZ SIMULINK Ó INTERFAZ MATLAB?.

En un principio se recurrió al uso del diagrama de bloques de Simulink para la implementar la animación.

Si en la ventana del diagrama de bloques se abre el menú desplegable de configuración y se elije un Solver con paso fijo, debido a que para el mismo paso el coste computacional de realizar la integración de las ecuaciones es aproximadamente el mismo se obtendrá una animación en la que la bicicleta no sufre aceleraciones o movimientos distintos en función del paso de integración escogido, lo que hace la simulación creíble con respecto al tiempo real que percibe el conductor.

El tiempo de la animación, al que podríamos llamar t' , se podría expresar como el tiempo real multiplicado por una constante que dependerá del paso de integración que escojamos, es decir:

$$t' = k \cdot t$$

El paso de integración se puede modificar de forma que el tiempo de la simulación se aproxime lo máximo posible al real.

Esta solución para la implementación de la simulación resulta muy burda debido a que nunca se tendrá un intervalo de tiempos de simulación igual al real y a que el paso de integración requerido depende de la computadora en la que se integre el modelo matemático, ya que el coste computacional de la integración en parámetros de tiempo es función del hardware disponible.

Inicialmente se adoptó esta solución en el proyecto, pero debido a los problemas antes mencionados se optó por implementar el modelo usando sólo el código de MATLAB, sin Simulink, para poder así obtener el tiempo del procesador mediante los comandos 'tic' 'toc'.

6.- ALGUNAS NOCIONES BÁSICAS SOBRE EL COMPORTAMIENTO DE LAS BICICLETAS.

Para la correcta interpretación de los resultados arrojados por el simulador es necesario tener una cierta comprensión acerca del comportamiento dinámico de una bicicleta.

Para una bicicleta típica, si se hace un estudio analítico de estabilidad en torno a la posición 'recta' se observa que esta solo es estable para un estrecho rango de velocidades en torno a 5m/s. Dicho estudio esta descrito en el documento de partida de este proyecto [2] y en él se muestra que para velocidades menores a 4.6m/s en la bicicleta del ejemplo existen autovalores con parte real positiva, siendo dos de los autovalores complejos en el caso de velocidades cercanas a 4.6m/s, lo que implica que la bicicleta será inestable y con comportamiento oscilante para velocidades menores.

Asimismo para velocidades superiores a 6.8m/s la parte real de uno de los autovalores aumenta lentamente hasta volverse positiva, teniéndose un comportamiento inestable pero fácil de compensar por parte del ciclista debido a que su valor es pequeño.

Los modos correspondientes a los autovalores anteriores son el *weaving*, el cual se caracteriza por el movimiento oscilante del manillar para pequeñas velocidades, el *castering*, dominado fuertemente por el giro del manillar, y el *wobble*, bamboleo de las ruedas típico de altas velocidades. Los tres aparecen situados encima de sus respectivos autovalores en la figura 10.

Estas inestabilidades se tendrán que ver reflejadas en la simulación.

Aunque el efecto giroscópico debido al giro de la rueda delantera tiene un efecto estabilizador, se puede demostrar fácilmente que el hecho de que la bicicleta sea estable para ciertas velocidades no es debido a este sino al ángulo de avance de la horquilla de la bicicleta. Llamaremos a dicho ángulo de avance β_0 . Una demostración de que el efecto giroscópico no es vital en la estabilidad de la bicicleta sería simplemente cambiar el valor de la masa del sólido 5 (rueda delantera) por cero en los parámetros del programa. Se observa que la bicicleta sigue siendo estable para un determinado rango de velocidades.

La distancia entre la intersección del eje de la horquilla con el suelo y el punto de contacto rueda-suelo también tiene una gran importancia dinámica. A esta distancia se le llamará en este texto por su denominación inglesa, '*trail*', y se representara con una c .

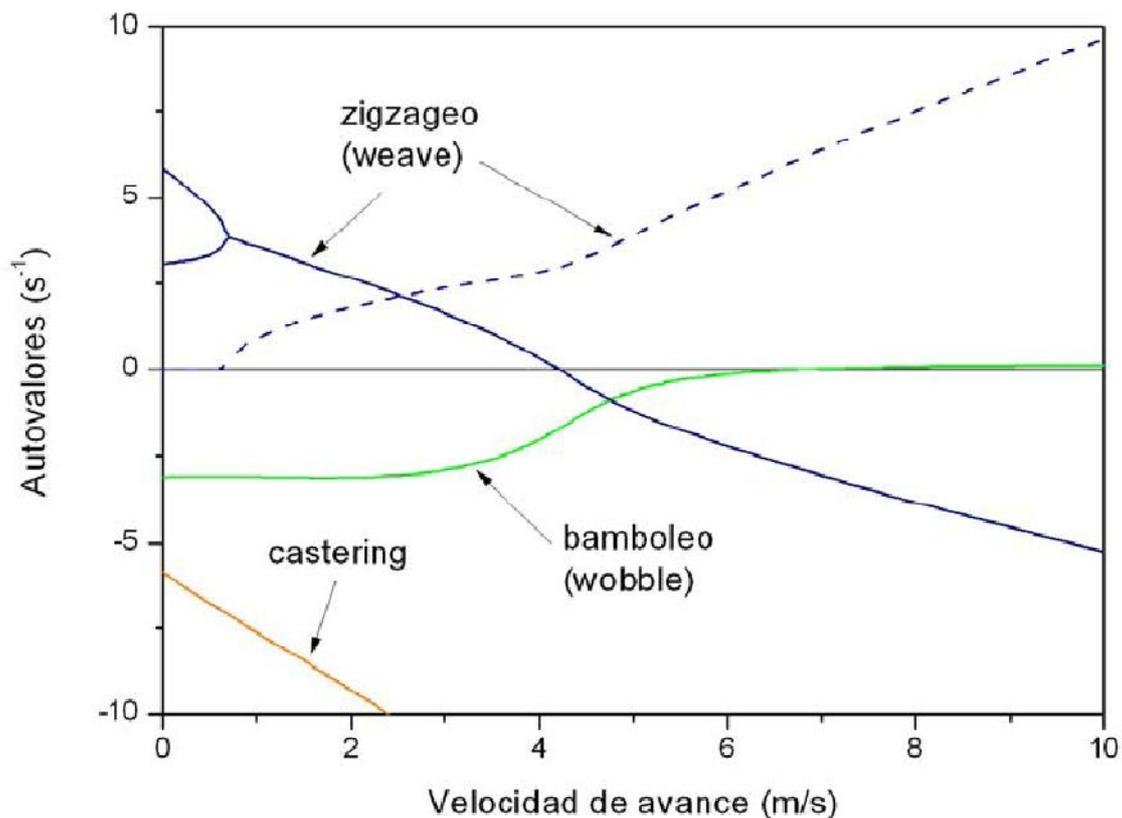


Figura 10

A continuación se va a deducir la relación entre el ángulo de vuelco θ y el giro del manillar γ debida al ángulo de avance de la horquilla de la bicicleta. Dicha relación es la que permite que la bicicleta sea autoestable para ciertas velocidades. En esta deducción se han realizado simplificaciones como las de que el giro del manillar es muy pequeño y se han despreciado efectos dinámicos como el giroscópico. Llamaremos b a la distancia entre puntos de contacto de las ruedas, ' a ' a la distancia longitudinal entre el punto de contacto de la rueda trasera y el centro de gravedad y ' h ' a la altura del centro de gravedad de la bicicleta más el cuerpo del ciclista. Los anteriores parámetros de la bicicleta se encuentran definidos gráficamente en la figura 11.

Como se aprecia en la figura 12, el radio de giro de la bicicleta para un γ pequeño puede calcularse como $R = \frac{b}{\gamma}$. (19)

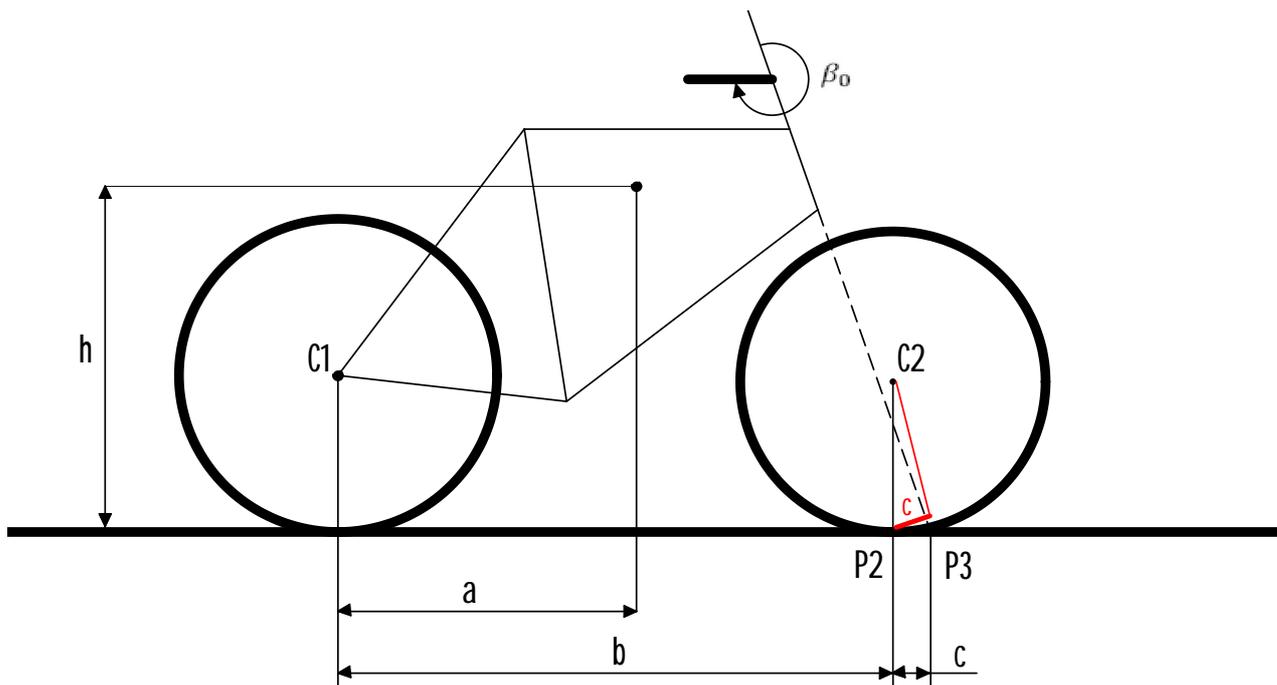


Figura 11

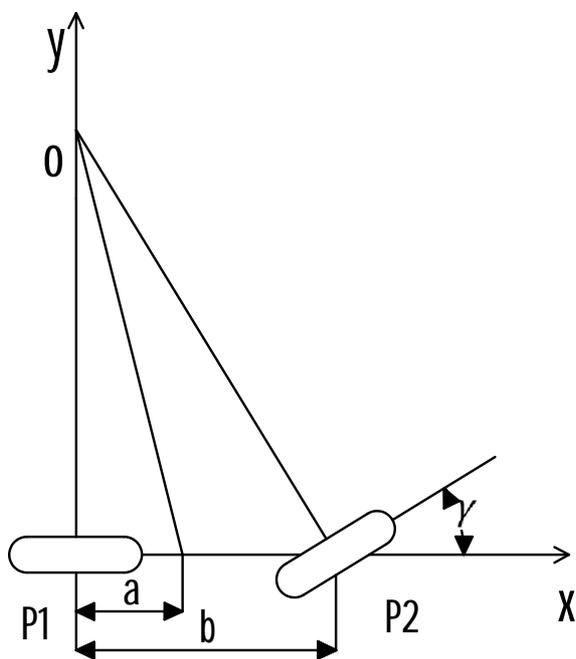


Figura 12

Al tener un cierto ángulo de avance la horquilla, los ángulos que relacionan la rueda delantera con el suelo no serán γ y θ , sino los resultantes al girar los ejes de coordenadas de la horquilla el ángulo β_0 , tal como se muestra en la figura 13.

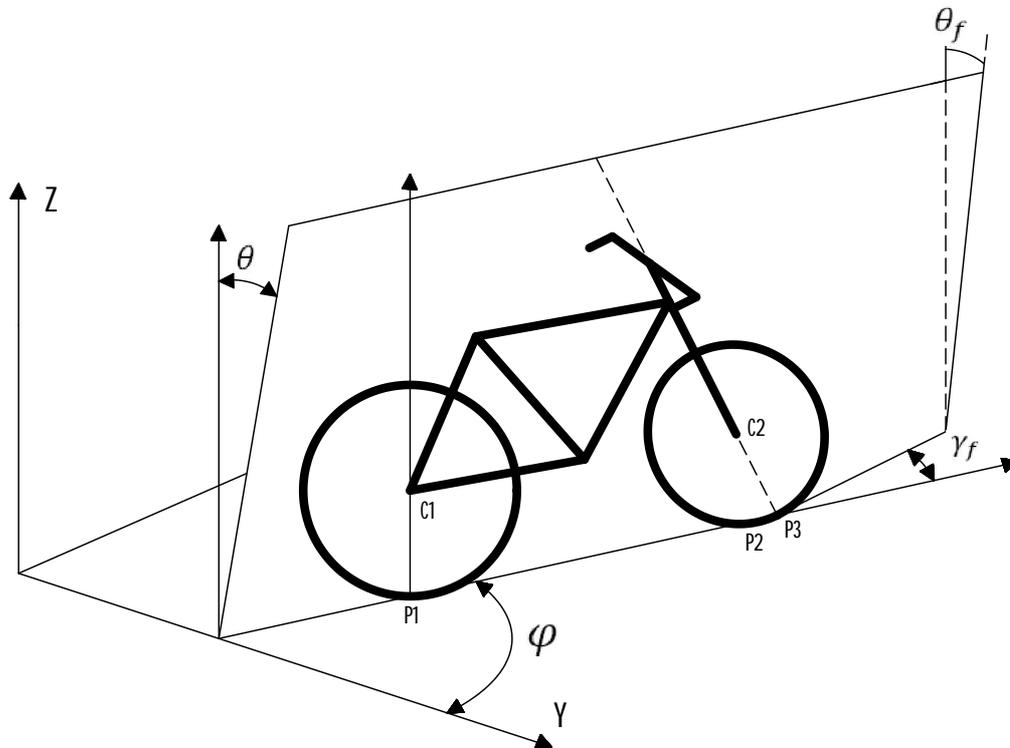


Figura 13

Dichos ángulos se llamarán respectivamente γ_f y θ_f , y su valor será:

$$\gamma_f = \sin \beta_0 \cdot \gamma \quad (20)$$

$$\theta_f = \theta - \cos \beta_0 \cdot \gamma \quad (21)$$

Calculando la fuerza centrífuga que actúa sobre la bicicleta como $F_c = \frac{mV^2}{R}$ y sustituyendo el valor de R quedará:

$$F_c = \frac{mV^2 \gamma_f}{b} \quad (22)$$

Haciendo un simple balance de momentos en el punto de contacto de la rueda trasera con el suelo podemos considerar que la fuerza perpendicular al plano de la rueda que se da en el contacto rueda delantera-bicicleta es:

$$F_f = \frac{mV^2 \gamma_f}{b} \frac{a}{b} = \frac{amV^2 \gamma_f}{b^2} \quad (23)$$

Y sustituyendo el valor de γ_f :

$$F_f = \frac{amV^2 \sin \beta_0 \cdot \gamma}{b^2} \quad (24)$$

En cuanto a la fuerza debida a la gravedad repitiendo un balance de momentos similar al hecho para la fuerza centrífuga quedará:

$$N_g = \frac{amg}{b} \quad (25)$$

Proyectando esta fuerza en el plano perpendicular al de la rueda delantera:

$$N_f = \frac{amg}{b} \sin \theta_f \simeq \frac{amg}{b} \theta_f \quad (26)$$

Si se multiplican estas dos fuerzas N_f y F_f por el brazo que las separa del eje de la horquilla y se hace un equilibrio de momentos en el eje de la horquilla se tendrá:

$$T = (F_f + N_f) \cdot d \quad (27)$$

$$d = c \cdot \sin \beta_0 \quad (28)$$

$$T = (F_f + N_f) \cdot c \cdot \sin \beta_0 \quad (29)$$

$$T = \frac{acm \sin \beta_0}{b^2} (V^2 \sin \beta_0 - bg \cos \beta_0) \cdot \gamma + \frac{acmg \sin \beta_0}{b} \cdot \theta \quad (30)$$

Donde T será el momento externo aplicado sobre el manillar. La ecuación anterior se puede expresar como la suma de dos coeficientes que multiplican a γ y a θ , en la cual si $V > \sqrt{bg \cot \beta_0}$ el coeficiente k_1 que multiplica a γ será positivo, es decir:

$$T = k_1 \gamma + k_2 \theta \quad \text{con } k_1 > 0 \quad \text{para } V > \sqrt{bg \cot \beta_0} \quad (31)$$

De esta ecuación se deduce que el giro del manillar de la bicicleta y la inclinación del marco están relacionados de forma que si surge algún tipo de momento debido a cualquier tipo de perturbación sobre el manillar se generará un par corrector debido al ángulo de avance proporcional al giro del manillar:

$$\theta = \frac{T_{\text{perturbación}} - k_1 \gamma}{k_2} \quad \text{con } k_1 > 0 \quad (32)$$

7.- CONTROL DE LA BICICLETA

Una vez realizada la escena en 3D y con una cierta comprensión acerca del comportamiento de las bicicletas se puede proceder a intentar controlar la bicicleta con distintas estrategias. En el presente proyecto han sido objeto de estudio el control imponiendo un ángulo en el manillar, el control aplicando un momento en el manillar y el control mediante la inclinación del cuerpo. A continuación se tratan dichas estrategias una por una.

7.1.- CONTROL IMPONIENDO EL ÁNGULO DEL MANILLAR.

En este caso el ángulo del manillar no será calculado integrando las ecuaciones sino que será una variable guiada cuyo valor vendrá determinado por la señal del joystick. También se calculará la velocidad y aceleración del giro del manillar dividiendo γ por el paso de integración en cada caso, lo cual se hará en la función Integrabicirc.m, declarando el ángulo del manillar y sus derivadas como variables globales con el fin de que puedan ser utilizadas por las funciones llamadas por Integrabicirc.m.

Como γ es una variable guiada no hace falta calcularla por lo que no hace falta una ecuación del sistema siguiente y se eliminará la décima :

$$\mathbf{M}(\mathbf{p})\ddot{\mathbf{p}} + \mathbf{D}^T\boldsymbol{\lambda} = \mathbf{Q}_v(\mathbf{p}, \dot{\mathbf{p}}) + \mathbf{Q}_{grav}(\mathbf{p}) + \mathbf{Q}_{ext} \quad (33)$$

$$\mathbf{C}(\mathbf{p}, \dot{\mathbf{p}}, t) = 0$$

A la matriz \mathbf{M} sin la décima ecuación se le llamará $\bar{\mathbf{M}}$ y será de 9x10. Si pasamos la tercera columna de dicha matriz, que solo multiplica a γ , al otro miembro de la ecuación nos queda:

$$\bar{\mathbf{M}}(\mathbf{p})\ddot{\mathbf{p}} + \mathbf{D}^T\boldsymbol{\lambda} = \mathbf{Q}_v(\mathbf{p}, \dot{\mathbf{p}}) + \mathbf{Q}_{grav}(\mathbf{p}) + \mathbf{Q}_{ext} - \mathbf{M}_3 \cdot \ddot{\gamma} \quad (34)$$

$$\mathbf{C}(\mathbf{p}, \dot{\mathbf{p}}, t) = 0$$

Donde $\bar{\mathbf{M}}$ ahora es de 9x9 y \mathbf{M}_3 se corresponde con la tercera columna de la matriz $\bar{\mathbf{M}}$ anterior. Se procede a convertir las ecuaciones anteriores en ODEs con el

método de *generalized coordinate partitioning* y ya se tienen las ecuaciones finales del sistema como en los dos métodos de control anteriores.

$$E^T \bar{M} E \ddot{p} = E^T (Q_v + Q_{grav} + Q_{ext} - M_3 \cdot \dot{\gamma}) \quad (35)$$

Cuando se intenta controlar la bicicleta imponiendo el ángulo el resultado que se obtiene es que al mínimo cambio en dicho ángulo la bicicleta cae irremediabilmente al suelo. Esto es debido a que el manillar ya no gira libremente hasta un ángulo que proporcione el momento que contrarreste el par creado por el peso del ciclista. Este resultado puede comprobarse fácilmente fijando en una bicicleta real el manillar en un cierto ángulo. Es evidente que dicha bicicleta resultará incontrolable y caerá.

Si se insiste en fijar cierto ángulo en el manillar también se observa que el simulador hace ‘raros’. Por ejemplo, la bicicleta se frena y comienza a andar para atrás. Esto es debido a que la inercia correspondiente al movimiento longitudinal de la bicicleta también participa en el equilibrio de momentos en el manillar y en una situación en la que el ángulo del mismo venga impuesto y la inclinación de la bicicleta no lo compense se podrá llegar a la situación en la que dicho movimiento longitudinal se invierta.

Usando las simplificaciones del apartado 6 y llamando \ddot{x} a la aceleración longitudinal de la bicicleta, con origen en el centro de masas del sólido bicicleta-ciclista:

$$F_{if} = F_i \sin \gamma_f = -m\ddot{x} \sin \gamma_f = -m\ddot{x} \sin (\sin \beta_0 \cdot \gamma) \simeq -m\ddot{x} \sin \beta_0 \cdot \gamma \quad (36)$$

Donde F_{if} es la componente de la inercia perpendicular al brazo d.

$$M_f = F_{if} \cdot d = F_{if} \cdot c \sin \lambda \beta_0 = -m\ddot{x} \sin \beta_0^2 c \gamma \quad (37)$$

Que se puede escribir como $M_f = -k_3 \gamma \ddot{x}$ $k_3 > 0$ (38)

Introduciendo la ecuación 16 en la ecuación 12’ queda:

$$T = k_1 \gamma + k_2 \theta - k_3 \gamma \ddot{x} \quad (39)$$

Se observa que cuando la bicicleta se está acelerando aparece un momento corrector con sentido contrario al giro del manillar, $T_{corrector} = -k_3 \gamma \ddot{x}$. De la misma manera, para un determinado γ , la inclinación de la bicicleta puede hacer que esta se decelere.

7.2.- CONTROL MEDIANTE UN MOMENTO EN EL EJE Z DEL MANILLAR.

Una vez desechada la idea de controlar la bicicleta imponiendo el ángulo del manillar es menester considerar el control aplicándole un momento. Aunque no es intuitivo, se puede observar que este tipo de control es el que realizamos sobre una bicicleta. Si se conduce con las palmas de la mano hacia arriba se tiene una mayor sensibilidad y se puede percibir que en realidad dejamos libre el manillar presionándolo levemente para corregir la dirección. Es por esto que los niños en proceso de aprendizaje suelen caerse de la bicicleta, ya que ante una situación de miedo se aferran al manillar imponiéndole un ángulo y no dejándolo libre.

Para introducir el par externo en las ecuaciones es necesario calcularlo en coordenadas generalizadas, para lo que se usa una nueva función ‘CalculaQmanillar.m’. Esta función recibe los valores del vector de coordenadas \mathbf{q} y calcula el momento generalizado \mathbf{M}_{mgen} , que será el siguiente:

$$\mathbf{M}_{mgen} \cdot \dot{\mathbf{q}} = \overline{\mathbf{M}_\gamma} \cdot \omega_{4l} \quad (40)$$

$$\omega_{4l} = \overline{\mathbf{G}_4} \cdot \dot{\mathbf{q}} \quad (41)$$

Sustituyendo (41) en (40):

$$\mathbf{M}_{mgen} \cdot \dot{\mathbf{q}} = [0 \ 0 \ M_\gamma] \cdot \overline{\mathbf{G}_4} \dot{\mathbf{q}} \quad (42)$$

$$\mathbf{M}_{mgen} = [0, 0, M_\gamma] \cdot \overline{\mathbf{G}_4} \quad (43)$$

La función vrjoystick de MATLAB toma como valores máximos correspondientes al movimiento del joystick los valores -1 y 1. Si los multiplicamos por cinco el momento máximo que se podría ejercer en el manillar será de 5Nm, lo cual es una cifra razonable.

Una vez calibrada la entrada del joystick ya se puede analizar la animación. Se comprueba que la bicicleta se maneja fácilmente y puede seguir sin problemas el circuito diseñado. El signo con el que se han definido los giros y momentos son de gran importancia y vienen resumidos en el cuadro siguiente descritos según la percepción que tendrá el ciclista. Por ejemplo, un giro del manillar en sentido horario tal y como vienen descritos los ejes de coordenadas del sólido 4 será un giro hacia la derecha.

	Joystick (Momento M_γ)	Leaning (Ángulo θ)	Steering (Ángulo γ)
Negativo	Izquierda	Izquierda	Derecha
Positivo	Derecha	Derecha	Izquierda

En el cuadro anterior se puede observar que el sentido de los ejes escogidos para el manillar no concuerda con el del joystick, ya que se busca que inclinar el joystick

hacia la izquierda sea lo mismo que hacer fuerza en el manillar hacia la izquierda, lo que se correspondería con un momento positivo en el mismo.

Para corregir esto se ha establecido que el momento aplicado en el manillar será la señal del joystick ' $u(1)$ ' cambiada de signo.

$$M_{\gamma} = -u(1)$$

Una vez hecho el cambio arriba indicado, cuando el joystick se inclina hacia la izquierda, la bicicleta gira a la derecha, ocurriendo lo mismo con los respectivos sentidos contrarios. Este hecho es revelador ya que para que la bicicleta gire a la derecha el ángulo del manillar tiene que ser negativo tal como viene descrito en el dibujo de la figura 1, y sin embargo el momento aplicado es en sentido positivo. Lo que realmente ocurre es que el manillar gira por un breve periodo de tiempo en el mismo sentido que el momento aplicado pero debido a que el cuadro de la bicicleta se posiciona en el sentido contrario aparece una fuerza creada por el ángulo de avance de la horquilla que hace que el manillar gire rápidamente hacia el mismo lado que la inclinación del cuadro. Este fenómeno se conoce como **counter steering** [3].

Usando la interfaz del simulador en Simulink nos es posible analizar el movimiento de la bicicleta ante distintos tipos de entradas. En la figura 14 se ha introducido como entrada una función escalón de 5Nm durante 0.05 segundos. Se puede ver claramente el fenómeno del counter steerig, ya que el ángulo del manillar comienza teniendo el mismo signo que el momento aplicado para después cambiar de signo, girando hacia la derecha, mismo sentido que el cuerpo de la bicicleta

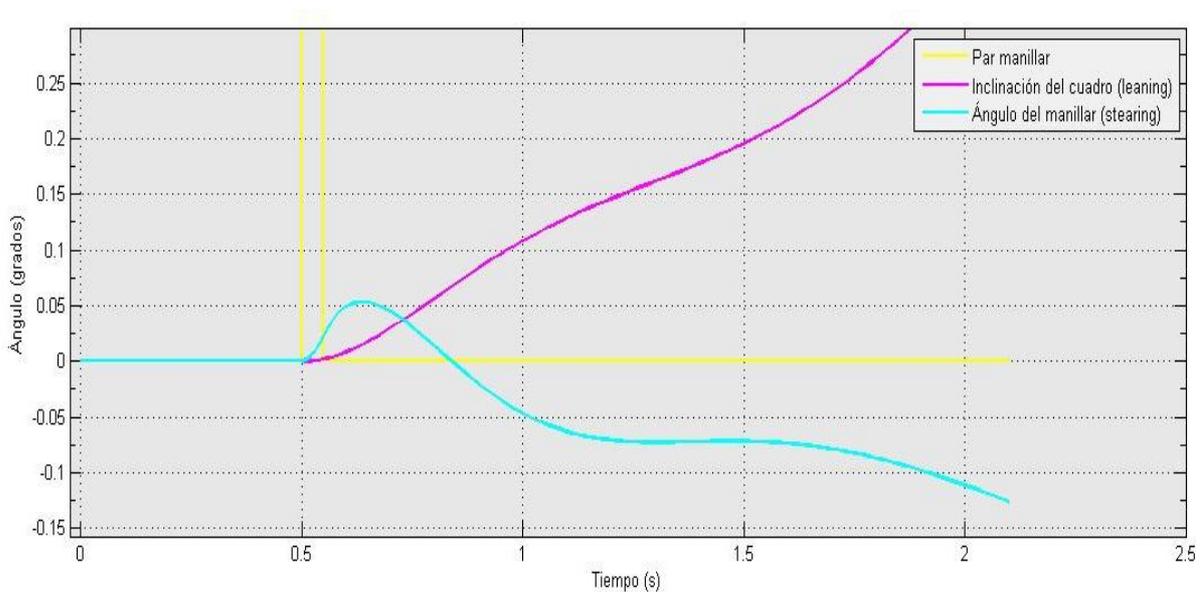


Figura 14

Es interesante observar que el cambio de signo en el ángulo del manillar se produce en apenas 0.3 segundos. Es tan repentino que por eso se tiene la impresión de hacer fuerza en el manillar en el mismo sentido que gira la bicicleta cuando se conduce una.

7.3.- CONTROL MEDIANTE LA INCLINACIÓN DEL CUERPO DEL CICLISTA.

Aunque se ha demostrado que una bicicleta puede ser dirigida mediante la aplicación de un momento en el manillar esta no es la única estrategia de control posible. Un control mediante el desplazamiento del centro de gravedad del conjunto bicicleta-ciclista podría explicar cómo un ciclista puede girar sin manos.

Se supondrá que el desplazamiento del centro de gravedad vendrá dado por el giro del tronco del ciclista con eje de giro en el contacto del mismo con el sillín de la bicicleta, por lo que se considerará el tronco como un sólido a parte, al que llamaremos sólido 6. Habrá que volver a calcular las ecuaciones del sistema para tener en cuenta este nuevo sólido.

Al nuevo ángulo correspondiente al giro del tronco del ciclista se le llamará τ . Los ejes de coordenadas de este nuevo sólido tendrán el centro en el contacto ciclista-sillín y la dirección resultante de girar los ejes del sólido 3 el ángulo τ . Definidos de esta manera los ejes de coordenadas, las matrices y los vectores correspondientes al sólido 6, que serán:

$$\mathbf{I}_6 = \begin{bmatrix} I_{x6} & 0 & I_{xz6} \\ 0 & I_{y6} & 0 \\ I_{xz6} & 0 & I_{z6} \end{bmatrix} \quad \mathbf{A}_\tau = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \tau & \sin \tau \\ 0 & -\sin \tau & \cos \tau \end{bmatrix}. \quad (38)$$

$\mathbf{A}_6 = \mathbf{A}_3 \cdot \mathbf{A}_\tau$ (Matriz de transformación de coordenadas locales a globales en sólido 6)

$$G_{6l} = \begin{bmatrix} \bar{0} \\ \bar{0} \\ A_6' \cdot k \\ (A_\theta \cdot A_\beta \cdot A_\tau)' \cdot i \\ \bar{0} \\ (A_\beta \cdot A_\tau)' \cdot j \\ \bar{0} \\ \bar{0} \\ \bar{0} \\ A_\tau' \cdot i \end{bmatrix}$$

Donde $\bar{0}$ es un vector nulo de 1x3 e i, j, k son los vectores unitarios.

Centro de masa en globales:

$$R_{g6} = R_{g2} + A_3 \cdot r_{sillin} + A_6 \cdot r_{g6} \quad (39)$$

Donde r_{sillin} y r_{g6} son respectivamente la posición del sillín y del centro de gravedad del sólido 6 en las coordenadas locales correspondientes a cada sólido. En el esquema de la figura 15 se muestra gráficamente como se han definido estos vectores.

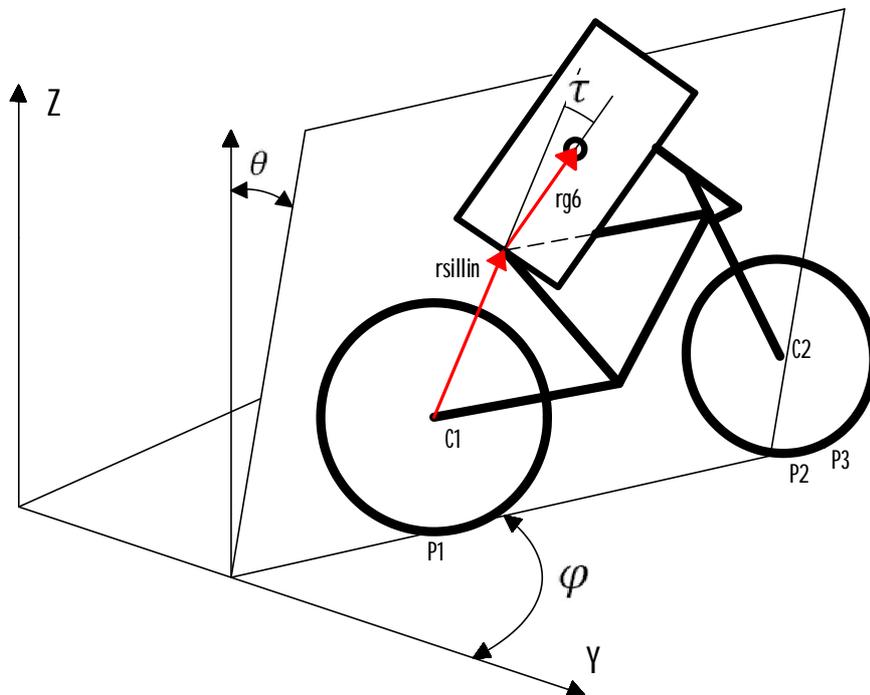


Figura 15

El resto de las matrices necesarias para las ecuaciones se calculan al igual que antes, usando el mismo programa de cálculo simbólico sin más que introducir en la ecuación de la energía cinética los términos correspondientes al nuevo sólido 6.

Ahora el sistema tiene 10 coordenadas contando τ , pero el mismo número de restricciones. Como τ es una variable guiada no hace falta calcularla con lo que no hace falta la décima ecuación del sistema siguiente:

$$\mathbf{M}(\mathbf{p})\ddot{\mathbf{p}} + \mathbf{D}^T\boldsymbol{\lambda} = \mathbf{Q}_v(\mathbf{p}, \dot{\mathbf{p}}) + \mathbf{Q}_{grav}(\mathbf{p}) + \mathbf{Q}_{ext} \quad (40)$$

$$\mathbf{C}(\mathbf{p}, \dot{\mathbf{p}}, t) = 0$$

A la matriz \mathbf{M} sin la décima ecuación se le llamará $\bar{\mathbf{M}}$ y será de 9x10. Si pasamos la decima columna de dicha matriz, que solo multiplica a τ , al otro miembro de la ecuación nos queda:

$$\bar{\mathbf{M}}(\mathbf{p})\ddot{\mathbf{p}} + \mathbf{D}^T\boldsymbol{\lambda} = \mathbf{Q}_v(\mathbf{p}, \dot{\mathbf{p}}) + \mathbf{Q}_{grav}(\mathbf{p}) + \mathbf{Q}_{ext} - \mathbf{M}_{10} \cdot \ddot{\tau} \quad (41)$$

$$\mathbf{C}(\mathbf{p}, \dot{\mathbf{p}}, t) = 0$$

Donde $\bar{\mathbf{M}}$ ahora es de 9x9 y \mathbf{M}_{10} se corresponde con la decima columna de la matriz $\bar{\mathbf{M}}$ anterior. Se procede a convertir las ecuaciones anteriores en ODEs con el método de *generalized coordinate partitioning* y ya se tienen las ecuaciones finales del sistema como en los dos métodos de control anteriores.

$$\mathbf{E}^T \bar{\mathbf{M}} \mathbf{E} \ddot{\mathbf{p}} = \mathbf{E}^T (\mathbf{Q}_v + \mathbf{Q}_{grav} + \mathbf{Q}_{ext} - \mathbf{M}_{10} \cdot \ddot{\tau}) \quad (42)$$

7.3.1.- CÁLCULO DE LOS PARÁMETROS DEL CUERPO DEL CICLISTA.

El cálculo de los parámetros del cuerpo humano supone un problema de gran complejidad el cual no se pretende abordar en este proyecto. Para realizar una correcta simulación bastará con unos valores aproximados del tensor de inercia y centro de gravedad del torso del ciclista.

Para obtener estos parámetros se ha supuesto el torso del ciclista como un prisma rectangular con dimensiones $L=0.8\text{m}$, $a=0.5\text{m}$ y $b=0.15\text{m}$. Si el torso tiene una masa de $m=50\text{kg}$ y se supone que esta está uniformemente distribuida, el tensor de inercia para el centro de gravedad, por ser las tres direcciones direcciones principales, será el siguiente:

$$\mathbf{I}_G = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \quad (43)$$

$$I_x = \frac{m}{12} (a^2 + L^2) = 3.708 \text{ kgm}^2$$

$$I_y = \frac{m}{12} (b^2 + L^2) = 2.76 \text{ kgm}^2$$

$$I_z = \frac{m}{12} (a^2 + b^2) = 1.135 \text{ kgm}^2$$

$$\mathbf{r}_g = \begin{bmatrix} 0 \\ 0 \\ z_g \end{bmatrix}$$

$$z_g = 0.4m$$

El sistema de coordenadas locales del sólido 6 tiene los ejes girados con respecto al del sólido 3 el ángulo β , que para la bicicleta en su posición vertical tiene el valor β_0 . Se quiere que el ciclista vaya en una posición erguida, con el tronco perpendicular al plano del suelo, que es la postura que los ciclistas suelen adoptar al conducir sin manos. Por ello habrá que deshacer el giro β_0 para que el centro de gravedad y el tensor de inercia en locales del sólido 6 sean los correspondientes a un conductor erguido.

$$\mathbf{A}_{\beta_0} = \begin{bmatrix} \cos \beta_0 & 0 & \sin \beta_0 \\ 0 & 1 & 0 \\ -\sin \beta_0 & 0 & \cos \beta_0 \end{bmatrix}, \quad \beta_0 = 5.9682 \text{ rad}$$

$$\mathbf{I}_{Grecto} = \mathbf{A}_{\beta_0}^T \cdot \mathbf{I}_G \cdot \mathbf{A}_{\beta_0} = \begin{bmatrix} 3.4610 & 0 & 0.7579 \\ 0 & 2.7600 & 0 \\ 0.7579 & 0 & 1.3820 \end{bmatrix} \text{ kgm}^2 \quad (44)$$

$$\mathbf{r}_{grecto} = \begin{bmatrix} 0.1239 \\ 0 \\ 0.3803 \end{bmatrix} m$$

Estos valores son los que finalmente se introducirán en las ecuaciones como parámetros del sistema. Para ver de qué manera afecta la inclinación del cuerpo al control de la bicicleta se podrá girar el cuerpo distintos ángulos para conseguir los correspondientes parámetros.

7.3.2.-ACERCA DEL PROGRAMA.

Al igual que en el programa para simular la bicicleta imponiendo el ángulo en el manillar, será necesario calcular las aceleraciones y derivadas del ángulo que viene impuesto, calculo el cuál se hará en 'Integrabici.m' declarando dichas variables como variables globales del programa para poder ser obtenidas en 'DerivadasBici.m'. De esta forma las derivadas de τ son iguales para cada paso de integración.

Si las derivadas de τ se calculasen en la función DerivadasBici.m el programa daría un error ya que el algoritmo Runge-Kutta evalúa la función DerivadasBici.m dos veces en el mismo instante t , con lo que $\Delta t = 0$ y $\dot{\tau} = \frac{\Delta\tau}{\Delta t} = \infty$.

El cálculo de la inercia asociada al ángulo τ se ha realizado mediante una función nueva llamada 'CalculaQcuerpo.m'.

Es importante remarcar que para este tipo de control lo cercano que este el tiempo de la animación al tiempo real es un factor crítico, debido a que la masa del torso del ciclista es muy grande en relación con la del sistema y cualquier discrepancia entre los tiempos real y de animación se expresa en unos cambios en las derivadas de τ que harán la bicicleta incontrolable debido a la elevada fuerza de inercia asociada al giro del cuerpo. Una animación a la mitad de velocidad que el tiempo real hará que la inercia del torso del ciclista sea cuatro veces mayor. Esto hace completamente inviable la implementación mediante el ajuste del paso de integración en Simulink.

7.3.3.- INTERPRETACIÓN DE LOS RESULTADOS.

Al ejecutar la simulación con la inclinación lateral del cuerpo como entrada se comprueba que es posible seguir el circuito, si bien la bicicleta responde de una forma más imprecisa a lo que el conductor desea hacer. La bicicleta sobrevira y para ciertas inclinaciones del cuadro es necesaria una inclinación del cuerpo tan grande para situar el centro de gravedad al otro lado del plano vertical al contacto de las dos ruedas con el suelo (y poder así levantar la bicicleta) que es imposible conseguirlo en un corto periodo de tiempo sin que el efecto de la inercia del cuerpo precipite la bicicleta al suelo.

Es posible dirigir la bicicleta de esta forma gracias al ángulo de avance de la horquilla, que hace que el manillar siga al cuadro y genere un par estabilizador tal y como se explicó en el capítulo 6. Para bicicletas con poco ángulo de avance la respuesta será tan lenta que la inclinación acumulada del cuadro será tal que la bicicleta caerá.

Estas observaciones se ajustan en gran medida a lo que ocurre cuando se intenta controlar una bicicleta sin manos, aunque seguramente lo que haga el ciclista no sea girar el cuerpo sino desplazar el centro de gravedad mediante el movimiento de las rodillas, el desplazamiento del apoyo con el sillín o de cualquier otra forma más difícil de modelar.

Por último llama la atención el escaso ángulo de giro del cuerpo necesario para provocar un giro en la bicicleta. Esto es debido a que en la bicicleta la relación entre el

peso del conductor y el del vehículo es muy elevada, de manera que el ángulo máximo del cuerpo considerado en la simulación es de entorno a 1 ó 2 grados. Esto no sería posible para otro tipo de vehículos, como por ejemplo una motocicleta.

7.4.-CONTROL COMBINADO: MOMENTO EN EL MANILLAR Y GIRO DEL CUERPO.

Aunque sólo con el momento en el manillar basta para conducir aceptablemente la bicicleta, el giro del cuerpo constituye un recurso que permite un mayor control y que podría ser utilizado por ciclistas con un relativo dominio.

En la versión final del simulador se han incluido sendas entradas. Mediante un mando con dos joysticks, con el izquierdo se puede introducir el momento en el manillar y en la rueda trasera y con el derecho se puede manejar la inclinación del cuerpo, tal como se ve en la figura 16 del final de este capítulo. Esto permite hacer estrategias como la siguiente:

A partir de un cierto radio de giro, la bicicleta tiene tendencia a caer hacia el lado hacia el cual está girando. Si se quiere seguir conservando dicho radio de giro se puede aplicar un momento en el manillar hacia el mismo sentido en el que la bicicleta se inclina, manteniendo así el ángulo θ constante. Otra forma de conseguir lo mismo sería inclinar el cuerpo hacia el lado contrario al giro para hacer que el momento debido al peso y a la aceleración centrífuga se equilibren. La diferencia entre sendos métodos de conservar el giro es que si se quiere súbitamente girar al lado contrario haciendo un zigzag, con el cuerpo inclinado hacia el lado contrario el cuadro se inclinará al lado contrario más rápido resultando más fácil la maniobra.

En opinión del redactor de este documento esta secuencia de acciones es la que realiza un ciclista cuando va a realizar giros muy cerrados, ya que además de poder cambiar la dirección de la bicicleta con más facilidad, hay que tener en cuenta que debido al sentido del equilibrio el conductor intenta mantener a verticalidad de su cuerpo ante una inclinación excesiva del cuadro, la cual da una sensación de que la bicicleta va a caer.



Figura 16

8.- ANÁLISIS MEDIANTE LA SIMULACIÓN DEL COMPORTAMIENTO DE LA BICICLETA.

Haciendo uso del simulador descrito podemos analizar el comportamiento de la bicicleta en algunos casos característicos e importantes para comprender la dinámica de esta.

8.1.-BICICLETA A BAJAS VELOCIDADES.

Si la velocidad de la bicicleta baja lo suficiente se puede apreciar como al entrar en el rango de velocidades para el cual es inestable empieza a zigzaguear hasta que cae. Este modo, conocido como weave, es como ya se dijo el correspondiente a dos autovalores complejos con parte real positiva en el análisis de estabilidad entorno al equilibrio vertical de la bicicleta. Dicha inestabilidad se puede corregir con la introducción de un control proporcional, que es además el tipo de control intuitivo que realizan los humanos.

Si hacemos que la inclinación del cuerpo sea siempre contraria a la del cuadro con un determinado $k > 1$, $\tau = -k * \theta$, se observa que la bicicleta no cae hasta una velocidad inferior a la habitual sin dicho control, y que cuando lo hace no cae con el característico movimiento en zigzag, sino que lo hace cayendo recta al suelo. Esto se debe a que se llega a la velocidad para la cual los dos autovalores antes mencionados dejan de ser complejos.

8.2.- DISTINTOS ÁNGULOS DE AVANCE PARA LA HORQUILLA DE LA BICICLETA.

Como ya se demostró, el ángulo de avance de la horquilla tiene una importancia capital en la dinámica de la bicicleta. Es por ello que resulta de gran interés el ver cómo se comporta para distintos valores.

En las ecuaciones de la bicicleta, tal y como viene descrita matemáticamente, dicho ángulo es el valor que toma β en la posición inicial de la bicicleta, al que hemos llamado previamente β_0 . Este valor no se introduce como un parámetro más sino que es calculado mediante un algoritmo de Newton-Raphson, que itera usando las restricciones holónomas (contacto y movilidad) y su jacobiano con respecto a β y ζ hasta que el valor de estas restricciones es el correcto (cero para la de contacto y Vt para la de movilidad).

Que el valor del ángulo de la horquilla sea el deseado depende de que los centros de gravedad en locales de los sólidos 2,4 y 5 tengan el valor para el que el algoritmo arroje como resultado el β_0 adecuado en la posición inicial del sistema, $\mathbf{q} = [0 \ 0 \ 0 \ 0 \ \beta_0 \ 0 \ 0 \ \zeta_0]$. Como sólo se desea colocar la horquilla en un ángulo distinto, la posición relativa de la rueda delantera con respecto a la horquilla se va a mantener constante (centro de gravedad de 5 en locales, $rG5=Cte.$) El valor que cambiará será el centro de gravedad del sólido 4, es decir, de la horquilla.

A partir de la restricción de contacto en la rueda delantera podemos obtener el valor de la componente vertical del vector entre el centro de gravedad del sólido 4 y el centro de gravedad del sólido 2 en globales ($RG2_z$), componente a la que llamaremos $rG4_{zglobales}$.

$$rG4_{zglobales} + RG2_z = -Restricción_z \quad (45)$$

$$Restricción = A_4 \cdot rG5 + A_5 \cdot \begin{bmatrix} R_d \cdot \cos \zeta \\ 0 \\ -R_d \cdot \sin \zeta \end{bmatrix}$$

Donde 'Restricción' es el vector en globales que une el c.d.g. del sólido 4 con el contacto suelo-rueda delantera.

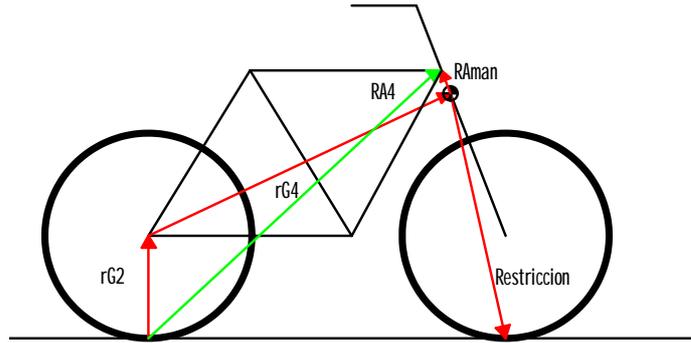


Figura 17

La componente horizontal del vector entre el centro de gravedad del sólido 4 y el centro de gravedad del sólido 2 en globales, a la que llamaremos $rG4_{xglobales}$, la podemos obtener a partir de la constancia del módulo del vector posición en coordenadas globales de la unión del marco con la horquilla para el instante inicial, vector al que llamaremos RA_4 . Esta unión está separada 2 centímetros del c.d.g. de la horquilla.

$$rG4_{xglobales} = \sqrt{|RA_4|^2 - (RG2_z + rG4_{zglobales} + RAman_z)^2} - RAman_x - RG2_x \quad (46)$$

$$RAman = A_4 \cdot \begin{bmatrix} 0 \\ 0 \\ 0.2 \end{bmatrix} \quad (47)$$

$$RA_4 = RG_4 + RAman \quad (48)$$

Ahora que tenemos las dos componentes no nulas del vector que une los centros de gravedad de los sólidos 2 y 4 en globales, podemos obtener $rG4$ transformando dicho vector a las coordenadas locales del sólido 4.

$$rG4 = A_3^T \cdot \begin{bmatrix} rG4_{xglobales} \\ 0 \\ rG4_{zglobales} \end{bmatrix}$$

La resolución de las anteriores ecuaciones para llegar al resultado que figura arriba se realizará con un programa de cálculo simbólico en MATLAB, tal y como se hizo para obtener las matrices y vectores de fuerza de las ecuaciones de Lagrange correspondientes a la bicicleta.

El resultado numérico de **rG4** se obtendrá con una nueva función llamada 'CalcularG4.m', en la que se sustituirán los parámetros y los valores de las coordenadas en el instante inicial en las ecuaciones resultantes del cálculo simbólico. Dicha función será llamada por 'Integrabicirkc.m' al principio del programa, de manera que al ejecutarse se pedirá al usuario el ángulo de avance deseado. Una vez introducido 'CalcularG4.m' devolverá a 'Integrabicirkc.m' los valores en coordenadas locales del centro de gravedad del sólido 4 y el resto del programa se ejecutará como en casos anteriores pero para estos nuevos valores.

Pudiendo simular la bicicleta para distintos ángulos de avance, se comprende mejor como afecta este parámetro a la dinámica de la misma. Para ello se han realizado una serie de gráficas de los autovalores de la bicicleta para distintas velocidades, los cuales se obtienen con el programa 'Calculaestabilidad.m', previo a la elaboración de este proyecto. 'Calculaestabilidad.m' no hace otra cosa que linealizar las ecuaciones del movimiento entorno a la posición vertical de la bicicleta para poder calcular las matrices de rigidez, de amortiguamiento y de masa y de ellas sus autovalores para cada velocidad longitudinal. La influencia del ángulo de avance se explicará mediante los siguientes casos representativos.

8.2.1.- Ángulo de avance menor que el inicial.

Si se comparan las gráficas de los autovalores de la bicicleta entorno a la posición vertical para un ángulo de avance de 5.9682 rad (el original) y otro de 5.75 rad se comprueba que dichas gráficas son prácticamente iguales, siendo la bicicleta estable en aproximadamente el mismo rango de velocidades. Sin embargo al ejecutar la animación se puede observar que la bicicleta es muy difícil de controlar con 5.75 radianes debido a que en una situación lo suficientemente apartada de la posición vertical el momento en el marco de la bicicleta generado por el giro del manillar aumenta muchísimo con poco que disminuya el ángulo de avance y la bicicleta gira mucho más de lo deseado.

En las figuras 17 y 18 de la siguiente página se pueden observar las gráficas de los autovalores con respecto a la velocidad.

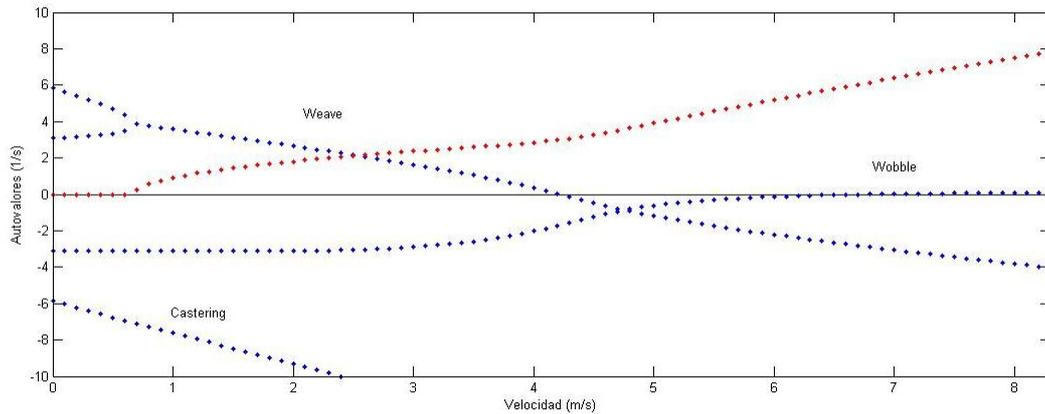


Figura 18.: Estabilidad para 5.9682 rad

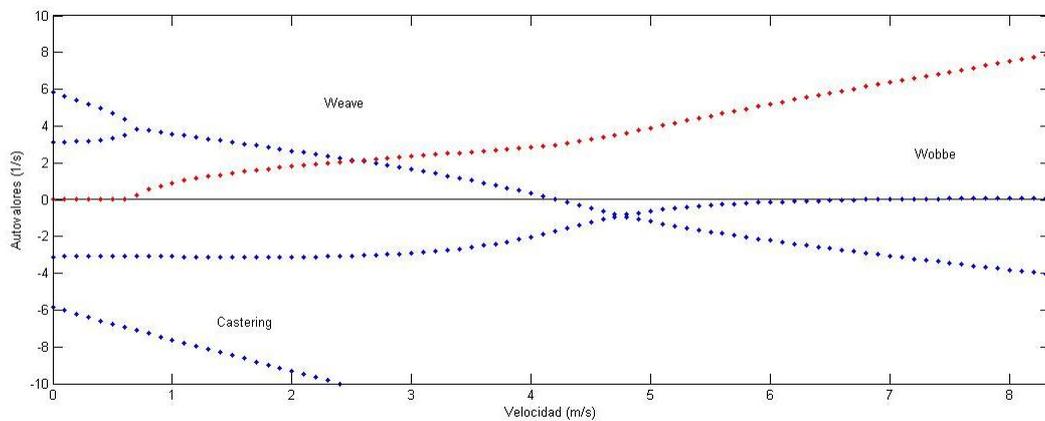


Figura 19.: Estabilidad para 5.75 rad

Se puede decir que a partir de un β_0 , la escasa mejora en la estabilidad de la bicicleta debida a una disminución del mismo no compensa la pérdida en maniobrabilidad que conlleva.

Para visualizar dicha pérdida de maniobrabilidad en este proyecto se ha tenido en cuenta un ratio al que se ha llamado ‘sensibilidad del cuadro’ y que no es otra cosa que el coeficiente entre el momento en el manillar realizado por el ciclista y la aceleración $\ddot{\theta}$ que experimenta el cuadro.

Adaptando la función ‘DerivadasBici.m’ se puede obtener una nueva función que para unos determinados parámetros calcule las aceleraciones $\ddot{\theta}$ posibles en función de los ángulos de avance para un momento de 1Nm en el manillar. A esta función se le ha llamado ‘Sensibilidad.m’.

‘Sensibilidad.m’ plotea la siguiente gráfica (figura 19) para el caso de la posición vertical de la bicicleta y la velocidad de 5m/s. En el eje vertical se representa $\frac{\ddot{\theta}}{M_{manillar}}$ y en el horizontal el ángulo de avance β_0 en radianes.

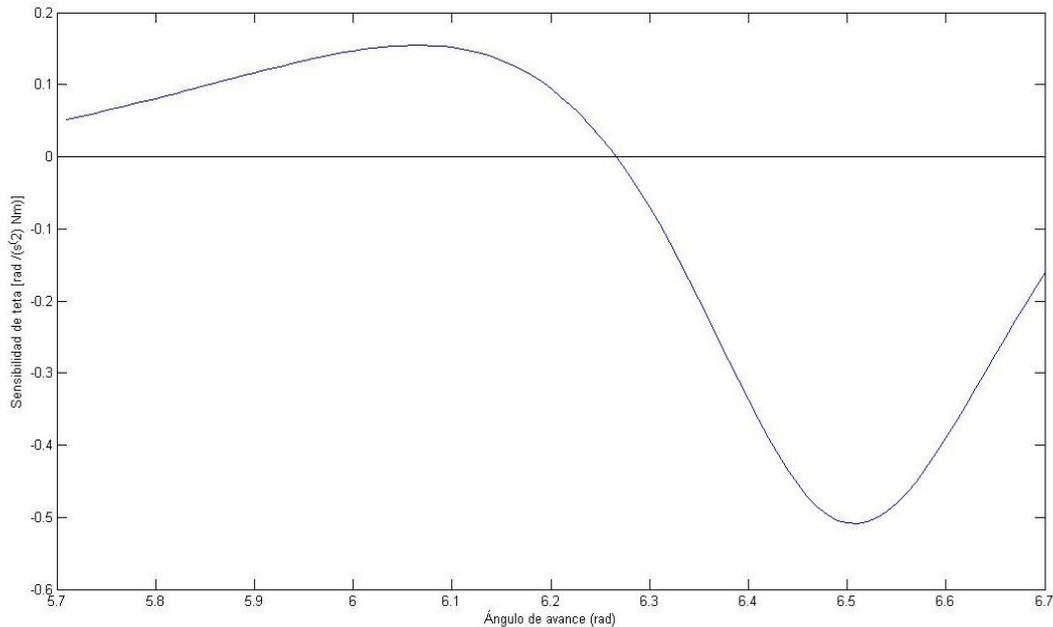


Figura 20.: Sensibilidad del cuadro con respecto al ángulo de avance

En la gráfica se observan varios hechos importantes. Se puede ver como para un valor del ángulo aproximado a 2π , el cual se corresponde con el ángulo de avance nulo, la aceleración pasa a ser de signo contrario al momento en el manillar, lo que significa que para los sistemas de coordenadas adoptados el cuadro de la bicicleta se movería en el mismo sentido que el giro aplicado en el manillar.

Por otro lado se aprecia que para ángulos menores a 6 cada se va reduciendo el valor de la sensibilidad de forma progresiva. Esto explica la disminución de la maniobrabilidad, ya que para que la bicicleta gire el manillar tiene que estar girado un cierto ángulo, lo que genera un momento en el cuadro que es compensado por el peso del ciclista, para lo cual el cuadro tiene que estar inclinado al mismo lado que la bicicleta gira. Si esta inclinación no es la suficiente para compensar el momento generado por el manillar, la bicicleta se enderezará y el manillar le seguirá dando una especie de ‘bandazo’ al lado contrario para valores del ángulo de avance excesivos.

Lo antes mencionado es lo que ocurre para 5.75 rad con respecto a 5.9682. Por ello, aunque la bicicleta es más estable, resulta menos maniobrable.

8.2.2.- Ángulo de avance mayor que el inicial.

Para ángulos bastante mayores que el inicial, cuando el valor de estos se acerca a 2π , la bicicleta también resulta incontrolable. No solo no se puede maniobrar con ella sino que además es inestable, como se observa en la figura 20, en la que se representan los valores de los autovalores para $\beta_0 = 2\pi$. Se aprecia que existen autovalores con parte real positiva para todo el rango de velocidades, además de haberlos complejos para también todo el rango.

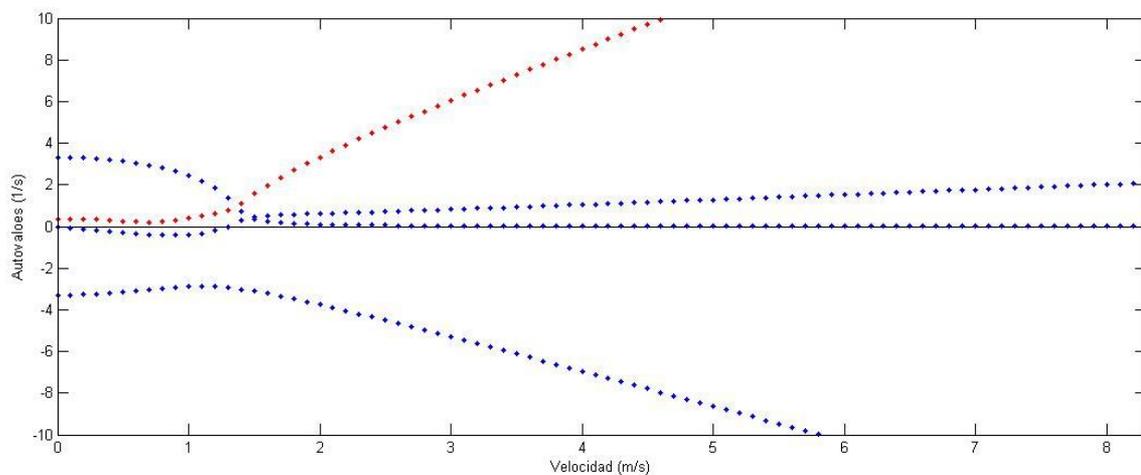


Figura 21

9.- CONCLUSIONES

Pese a lo que inicialmente parecía, se puede controlar aceptablemente la simulación de una bicicleta. Aunque el manejo de la bicicleta nunca será tan bueno como en la realidad debido a la falta de información sensorial, se puede seguir sin problemas un circuito con distintos tipos de curvas. La clave para ello es la simulación en tiempo real, ya que sin ella el desfase entre lo que hace el usuario y lo que percibe imposibilita la conducción.

El cuerpo del ciclista, aunque útil a la hora de manejar la bicicleta, no es necesario, pudiéndose controlar esta solo con la aplicación de momentos en el manillar y permaneciendo el cuerpo siempre en el plano del cuadro.

También se ha comprobado la importancia relativa de los parámetros de la bicicleta, en especial la del ángulo de avance, del cual depende en gran medida la dinámica de la misma. Un ángulo de avance positivo es imprescindible para que esta pueda ser controlada sin ningún artificio, aunque si la inclinación de la horquilla resulta excesiva, se dificulta la capacidad de maniobra.

El comportamiento de la simulación ha resultado ser bastante verosímil y similar al de una real, lo que pone de manifiesto la utilidad de un simulador bien implementado como una herramienta para el análisis físico de vehículos.

10.- MEJORAS FUTURAS.

El autor del proyecto considera que en un futuro otros alumnos podrán realizar mejoras sobre este que escapan al presente trabajo, como son las siguientes.:

- Si el código se programase en C++ se prescindiría del MATLAB como intérprete del mismo y se podrían usar funciones multitarea lo que mejoraría la velocidad de cálculo del simulador. Se sugiere el uso de la 'toolbox' de MATLAB 'Real Time Workshop', con la cual se puede compilar en C a partir del código de MATLAB y el joystick puede ser introducido como una tarjeta de adquisición.
- El modelo matemático de la bicicleta se encuentra muy simplificado. Sería interesante en un futuro que se tuvieran en cuenta aspectos como la suspensión de la bicicleta y un modelo elástico de neumático, que según algunas publicaciones tienen importancia en el comportamiento de la misma. También se podrían considerar planos inclinados y otros perfiles como suelo y situar los frenos en las respectivas ruedas, no como un momento en contra en los pedales.
- La percepción del usuario del simulador es mejorable de distintas formas. Por ejemplo, en este proyecto se ha probado usar un volante con force feedback, que está soportado por la 'toolbox' de realidad virtual de MATLAB. Se abandonó esta línea de mejora por problemas en la calibración del mismo que se antojan solucionables a largo plazo.

11.- DESCRIPCIÓN DE LOS ARCHIVOS ADJUNTOS:

Adjuntos en un CD con el presente trabajo se facilitan una serie de archivos, todos ellos en la carpeta de '**Programa**'. Entre dichos archivos se encuentran las versiones del simulador objeto de este proyecto, por lo que es menester una descripción detallada de los mismos.

TIPOS DE ARCHIVO:

En la carpeta 'Programa' se pueden encontrar los siguientes tipos de archivo. Todos ellos deberían poder abrirse en un ordenador con un sistema operativo Windows XP, Windows Vista o Windows 7 en el que se encuentre instalado el programa MATLAB 2009 o superior con las Toolboxes correspondientes a la instalación por defecto.

- Archivos **.wrl** : Es el formato de las escenas implementadas en VRML. Hay al menos un archivo .wrl por cada versión del simulador.

- Archivos **.m**: Estos archivos contienen el código de los programas de MATLAB.

- Archivos **.mdl** : Se corresponden con los diagramas de bloques de Simulink.

- Archivos **.jpg**: Las texturas y fotos incluidas en el proyecto irán en este formato.

- Archivos **.txt**: En este formato se guardan los valores simbólicos de las matrices que intervienen en las ecuaciones.

CONTENIDO DE LA CARPETA PROGRAMA:

La carpeta Programa contiene una serie de subcarpetas, las cuales se describen a continuación:

1.- Programa definitivo: En esta carpeta se encuentran las distintas versiones del simulador final de la bicicleta. Estas versiones son las siguientes:

1.1.- Momento en el manillar.

1.2.- Inclinación del cuerpo y momento en el manillar.

1.3.- Giro del manillar.

1.4.- Pregunta ángulo de avance.

1.4.1.- Pregunta ángulo de avance con momento en el manillar.

1.4.2.- Pregunta ángulo de avance con inclinación del cuerpo y momento en el manillar.

Cada una de las tres primeras versiones se corresponde con un método de control diferente y la cuarta versión se corresponde con la modificación del programa para poder elegir el ángulo de avance al comenzar la simulación. Esta cuarta versión contiene a su vez dos subcarpetas, una para el control con momento en el manillar y otra para control con inclinación del cuerpo y momento en el manillar.

Para hacer funcionar la simulación, el usuario debe ejecutar en MATLAB el archivo 'Principal.m', tras lo cual se abrirá la escena. Es importante tener conectado el mando al ordenador antes de ejecutar el archivo, o de lo contrario MATLAB comunicará un error al no encontrar el periférico.

Si se desean probar una a una todas las versiones se recomienda comenzar en el mismo orden que las carpetas. Si se desea probar directamente la versión más completa se recomienda comenzar por la carpeta 1.4. Para que la animación comience en cualquiera de los programas de la carpeta 1.4 antes habrá que introducir el valor del ángulo de avance deseado en radianes, en el mismo sentido en el que se describe en la memoria. Este valor se introducirá en la ventana de comandos de MATLAB, en la cual aparecerá una línea de caracteres requiriendo dicho ángulo.

Cuando se ejecuta una simulación se recomienda dejar correr esta un par de segundos sin tocar el joystick, debido a que en ciertas ocasiones un 'bug' de origen desconocido hace que la bicicleta se precipite al suelo al recibir una señal del joystick en los primeros segundos de la animación.

2.- Varios: La otra carpeta dentro de Programa. En ella se encuentran carpetas con archivos auxiliares o de menor importancia dentro del proyecto.

2.1.- Versión de Simulink: En esta carpeta se encuentra una versión correspondiente al giro con momento en el manillar implementado en Simulink. Hay que tener en cuenta que dicha versión no es la definitiva, y se incluye porque es importante en cuanto a ilustrar el proceso seguido para llegar a la versión definitiva. En esta versión se acelera dejando pulsado el botón 1 y se decelera con el botón dos, siendo la función de cada joystick la misma que con la interfaz en MATLAB.

2.2.- Versión tentempié: En esta versión al simulador de la bicicleta se le ha introducido un control proporcional de manera que el cuerpo del ciclista contrarresta el ángulo del cuadro, con lo que la bicicleta a bajas velocidades se resiste a caer. El experimento es el comentado en el capítulo **8.1** y para hacerlo funcionar basta con ejecutar el archivo 'Principal.m' con MATLAB y una vez la animación este

funcionando, dar un pequeño toque al joystick para introducir una pequeña perturbación y que la bicicleta empiece a oscilar siguiendo el modo de weaving.

2.3.- Programas de cálculo simbólico: En esta carpeta se encuentran los programas de MATLAB que se han utilizado para el cálculo simbólico de las ecuaciones de la bicicleta con y sin inclinación del cuerpo, así como el programa utilizado para calcular $rG4$ en función del ángulo de avance.

12.- BIBLIOGRAFÍA

Para la elaboración de este proyecto el autor se ha apoyado en los siguientes textos :

- **“Dinámica de una bicicleta”**, de Jose Luis Escalona.
- **“Bicycles dynamics and control: Adapted bicycles for education and research”**, de Karl J. Åström.
- **“Steady-state handling characteristics of a bicycle”**, de Stephen M. Cain and Noel C. Perkins.
- **“Some recent developments in bicycle dynamics and Control”** de A. L. Schwab, J. D. G. Kooijman y J. P. Meijaard.

Mi más profundo agradecimiento a todos.

También han sido esenciales los manuales siguientes:

- **V-Realm™ Builder**
User's Guide and Reference.
- **Ayuda del MATLAB**, en especial la referente a la **Toolbox de Virtual Reality**.

REFERENCIAS

[1] MODELO DE WHIPPLE:

En 1899 el meteorólogo y matemático Francis Whipple, justo después de ser nombrado profesor del Trinity College (Cambridge), escribió su tesis sobre la dinámica de las bicicletas. El modelo matemático de bicicleta que en ella figuraba consistía en un sistema compuesto por 5 sólidos, véase, el suelo, las dos ruedas, el cuadro y el manillar. Las ecuaciones no lineales de dicho sistema eran casi correctas, pero en ellas faltaba un término ‘coseno de psi’ en una de las ecuaciones de restricción, error el cual se propagaba al resolver las ecuaciones no lineales. Afortunadamente este error no tenía influencia en las ecuaciones linealizadas. Whipple fue el primero en obtenerlas y estudió la estabilidad de la bicicleta mediante momentos aplicados en el manillar y la inclinación del cuerpo.

El modelo de bicicleta del presente proyecto está basado en el modelo de Whipple.

[3] Texturas: Son archivos de imagen utilizados por los programas de CAD y en VRML para dar apariencia a los sólidos adaptando la imagen a su superficie. Por problemas de compatibilidad el editor de VRML no acepta las texturas de la biblioteca de SolidWorks. Las texturas que se han usado en las escenas VRML son archivos .jpg y se han guardado en la carpeta ‘textures’ a la cual se hace referencia en dichas escenas.