

Trabajo Práctico 3

Viernes 20 de Junio de 2014

Métodos Numéricos

Aproximación de funciones mediante técnicas de Análisis Numérico.

Integrante	LU	Correo electrónico
Pedro Rodriguez	197/12	pedrorodriguezsjs@hotmail.com
Martín Caravario	470/12	martin.caravario@gmail.com
Esteban Fernandez	691/12	esteban.pmf@gmail.com



Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires
Ciudad Universitaria - (Pabellon I/Planta Baja)
Intendente Güiraldes 2160 - C1428EGA
Ciudad Autónoma de Buenos Aires - Rep. Argentina
Tel/Fax: (54 11) 4576-3359
<http://www.fcen.uba.ar>

Índice

1. Introducción	2
2. Desarrollo	2
3. Resultados	9
4. Discusión	19
5. Conclusiones	22
6. Apéndices	25
6.1. Enunciado	25
7. Referencias	29

Resumen

Desde hace ya varios años viene llevándose a cabo la competencia de fútbol de robots y es aquí donde diferentes equipos de todo el mundo demuestran sus destrezas a nivel intelectual para formar el equipo más inteligente capaz de vencer a cualquier adversario y así consagrarse campeones mundiales.

En esta oportunidad se nos asignó la tarea de crear la inteligencia artificial del arquero de manera tal que sea lo suficientemente intrépido para atrapar todos los tiros al arco posibles. Para esto lo que haremos será, en función del tiempo, ver como se moviliza la pelota sobre el campo de juego para, aprovechando esos datos observados, ir estimando instante a instante la función que describe la pelota. Una vez hecho esto intentaremos predecir, en base a la función obtenida, por donde pasará la pelota cuando llegue a la línea del arco y así mover al arquero convenientemente para que la atrape.

Palabras clave: Aproximación de funciones - Interpolación polinómica - Splines Cúbicos - Cuadrados Mínimos

1. Introducción

El presente informe tiene como finalidad introducir al lector, mediante una aplicación, a los conceptos de análisis numérico utilizados para la aproximación y estimación de funciones a partir de una muestra de datos observados. Para esta aplicación utilizaremos interpolación polinómica mediante Splines Cúbicos. A continuación se presentarán los fundamentos teóricos que crean las bases de nuestra implementación. Más adelante en este informe se entrará en detalle sobre este método implementado.

- Nuestra implementación utilizará la técnica de Interpolación Polinómica para la aproximación de funciones, en particular, mediante uno de los métodos de interpolación segmentaria conocido como Splines Cúbicos, el cual será desarrollado en la próxima sección de este informe. El mismo, a grandes rasgos, se encargará de segmentar el dominio (dado por los datos muestrales) de nuestra función a estimar, y por cada intervalo definido se creará un polinomio asociado, los cuales luego, con algunas condiciones que impondremos previamente a la hora de calcularlos, podremos unir y formar una única función partida que incluye a cada uno de estos que, intencionalmente mediante las condiciones que impondremos, pasará por todos los puntos observados de la muestra.

A modo informativo y como trabajo a futuro, en el final de las conclusiones se hablará de un método alternativo no implementado en código pero que puede ser utilizado también para resolver el problema planteado; se trata del método de Cuadrados Mínimos, el cual, enmarcado en la optimización matemática, dado un conjunto de puntos de una muestra observada y tomando una familia de funciones, se intenta encontrar una función continua dentro de esa familia que mejor se aproxime a los datos (es decir, que presenten un mejor ajuste) de acuerdo con el criterio de mínimo error cuadrático.

- En esta aplicación, en particular, se buscará mover el arquero instante a instante con diferentes criterios (dichos criterios serán desarrollados más adelante). El principal de ellos será, una vez obtenida la función estimada que estará en función del tiempo y a partir de esta, buscar el instante en el cual la pelota pasa por la línea del arco antes de que haya pasado realmente, es aquí donde se ve la noción de predicción y estimación. Dicho instante será encontrado apartir de la búsqueda de la raíz en esta función obtenida. A nosotros nos interesará entonces modificar levemente este algoritmo de búsqueda de raíces con el fin de obtener dicho instante, luego se explicará el por qué. Para obtenerlas, utilizaremos el método de la bisección que también será desarrollado y explicado más adelante en el presente informe.

2. Desarrollo

En el presente informe realizaremos experimentaciones las cuales evaluarán la efectividad del método implementado, se verá además como dicha efectividad afecta a los movimientos que realiza el arquero a lo largo del tiempo, entre otras cuestiones. Antes de procededr con la experimentación, es necesario realizar un desarrollo del procedimiento y de la teoría utilizada para implementar nuestra resolución del problema en el lenguaje de programación C++.

El uso de Splines Cúbicos para resolver el problema de la interpolación de puntos mediante polinomios deriva de los inconvenientes que presenta la interpolación polinómica ordianaria, la cual toma un único polinomio de grado elevado para interpolar todos los puntos (dicho grado depende de la cantidad de puntos a interpolar). En particular, cuando hay muchos puntos en nuestra muestra y se intenta interpolarlos considerando un único polinomio de grado elevado, el mismo presentará oscilaciones cada vez mayores que quitarán precisión a nuestra aproximación de la función descripta por los puntos analizados.

Mediante la técnica de Splines Cúbicos uno evita este inconveniente, ya que se limita el grado del polinomio. Para esto, se divide en subintervalos el intervalo a interpolar y se utiliza un polinomio por cada uno de ellos. Las condiciones de interpolaciones se satisfacerán localmente, es decir en cada uno de los polinomios utilizados para interpolar en cada uno de los subintervalos definidos a partir del intervalo.

Definición: Dados $n+1$ puntos $(x_0, y_0) \dots (x_n, y_n)$ tales que $x_0 < \dots < x_n$. En la interpolación mediante Splines Cubicos, utilizaremos polinomios de grados ≤ 3 para interpolar en cada intervalo $[x_{i-1}, x_i]$ y con ellos armar la función spline la cual notaremos $S(x)$. S tiene derivada de orden $k - 1$ continua en $[x_0, x_n]$, es decir, S es de clase $C^{k-1}([x_0, x_n])$. Entonces, S es un polinomio continuo a trozos de grado a lo sumo k (donde k es el grado del polinomio), que tiene derivadas continuas de hasta orden $k - 1$. S , será finalmente la unión de polinomios de grado menor o igual que tres que se unen de manera continua.

Dada la siguiente tabla de valores:

x	x_0	x_1	\dots	x_n
y	y_0	y_1	\dots	y_n

Se trata de construir un spline cúbico para interpolar dicha tabla. Para conseguir esto, es necesario encontrar los coeficientes que acompañan cada polinomio cúbico de cada subintervalo determinado.

El spline quedará definido de la siguiente manera:

$$S(x) = \begin{cases} S_0(x) = a_{0,3}x^3 + a_{0,2}x^2 + a_{0,1}x + a_{0,0} & x \in [x_0, x_1] \\ \vdots & \\ S_{i-1}(x) = a_{i-1,3}x^3 + a_{i-1,2}x^2 + a_{i-1,1}x + a_{i-1,0} & x \in [x_{i-1}, x_i] \\ S_i(x) = a_{i,3}x^3 + a_{i,2}x^2 + a_{i,1}x + a_{i,0} & x \in [x_i, x_{i+1}] \\ \vdots & \\ S_{n-1}(x) = a_{n-1,3}x^3 + a_{n-1,2}x^2 + a_{n-1,1}x + a_{n-1,0} & x \in [x_{n-1}, x_n] \end{cases}$$

Al haber n polinomios y 4 coeficientes a determinar por cada uno de ellos, entonces serán $4n$ incógnitas en total. Los polinomios S_{i-1} y S_i , para todo $i = 1 \dots n-1$ interpolan en el mismo valor en el punto x_i :

$$\begin{aligned} S_{i-1}(x_i) &= y_i \\ S_i(x_i) &= y_i \end{aligned}$$

Esto condicionará S de manera tal que la misma sea continua, dando lugar a $2(n-1) = 2n-2$ condiciones.

Además, $S_0(x_0) = y_0$ y $S_{n-1}(x_n) = y_n$, estas son dos condiciones más a cumplir.

La función derivada S' también debe ser continua:

$$S'_{i-1}(x_i) = S'_i(x_i) \text{ para } i = 1, 2, \dots, n-1$$

De aquí se desprenden otras $n-1$, análogamente impondremos la misma condición sobre la segunda derivada de S .

$$S''_{i-1}(x_i) = S''_i(x_i) \text{ para } i = 1, 2, \dots, n-1$$

Entonces, hasta el momento estaremos imponiendo en total: $(2n-2) + 2 + 2(n-1) = 4n-2$ condiciones que han de ser verificadas.

Se obtiene un sistema de $4n-2$ ecuaciones con $4n$ incógnitas. Se necesitan agregar entonces 2 condiciones más de manera tal que el sistema quede cuadrado y unívocamente determinado.

Para construir un spline cúbico interpolante en los puntos $x_0 < x_1 < \dots < x_n$, conviene expresar los polinomios cúbicos de la forma (lo cual luego simplificará la resolución del sistema resultante de la imposición de las condiciones nombradas previamente):

$$S_j(x) = a_j + b_j(x - x_j) + c_j(x - x_j)^2 + d_j(x - x_j)^3, \quad j = 0, 1, 2, \dots, n-1$$

Existen más de un tipo de Spline Cúbico, los tipos más comunes de splines utilizados son los siguientes (difieren en esas dos condiciones que nos faltan).

Spline cúbico amarrado o de frontera sujeta: Cuando se conoce el valor de la derivada primera en los extremos, o, al menos una aproximación de los mismos.

$$S'(x_0) = f'(x_0), S'(x_n) = f'(x_n)$$

Como esta condición está ligada a información acerca de la función que se intenta aproximar, se logran mejores resultados.

Spline cúbico natural de frontera libre: Cuando se conocen las derivadas segundas en los puntos inicial y final, o bien se toman:

$$S''(x_0) = S''(x_n) = 0.$$

Esto nos dará el resultado más suave posible, en el sentido de curvatura mínima. Hacia los extremos del intervalo de interpolación, la curva se convierte en una recta.

Un caso particular del spline natural es en el que se calculan las derivadas segundas en los extremos interpolando el valor que toman en los dos nodos más próximos, es decir:

$$\begin{aligned} S''(x_0) &= (1 + \frac{h_0}{h_1})S''(x_1) - \frac{h_0}{h_1}S''(x_2) \\ S''(x_n) &= (1 + \frac{h_{n-1}}{h_{n-2}})S''(x_{n-1}) - \frac{h_{n-1}}{h_{n-2}}S''(x_{n-2}) \end{aligned}$$

Finalmente, el cálculo de los coeficientes se realiza en dos pasos:

Se resuelve primero un sistema tridiagonal de $n-1$ ecuaciones con $n-1$ incógnitas, que son las derivadas segundas del spline en cada x_i . $S''(x_i) = r_i, i = 1, \dots, n-1$.

$$\begin{pmatrix} \frac{h_0+h_1}{3} & \frac{h_1}{6} & 0 & 0 & \dots & 0 & 0 \\ \frac{h_1}{6} & \frac{h_1+h_2}{3} & \frac{h_2}{6} & 0 & \dots & 0 & 0 \\ 0 & \frac{h_2}{6} & \frac{h_2+h_3}{3} & \frac{h_3}{6} & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \frac{h_{n-3}}{6} & \frac{h_{n-3}+h_{n-2}}{3} & \frac{h_{n-2}}{6} \\ 0 & 0 & 0 & 0 & \frac{h_{n-2}}{6} & \frac{h_{n-2}+h_{n-1}}{3} & \frac{h_{n-1}}{6} \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_{n-1} \end{pmatrix} = \begin{pmatrix} e_1 - e_0 - \frac{h_0}{6}r_0 \\ e_2 - e_1 \\ e_3 - e_2 \\ \vdots \\ e_{n-2} - e_{n-3} \\ e_{n-1} - e_{n-2} - \frac{h_{n-1}}{6}r_n \end{pmatrix}$$

Donde $h_i = x_{i+1} - x_i$, $e_i = \frac{y_{i+1} - y_i}{x_{i+1} - x_i}$, $i = 0, \dots, n-1$

Luego, una vez resuelto el sistema, si queremos calcular en un determinado x el valor del Spline Cúbico calculado deberemos tomar lo siguiente:

$S_i(x) = a_i(x - x_i)^3 + b_i(x - x_i)^2 + c_i(x - x_i) + d_i$, $x \in [x_i, x_{i+1}]$, $i = 0, \dots, n-1$. Se verifica que:

$$\begin{aligned} a_i &= \frac{r_{i+1} - r_i}{6h_i} \\ b_i &= \frac{r_i}{2} \\ c_i &= \frac{y_{i+1} - y_i}{h_i} - \left(\frac{r_{i+1}}{6} + \frac{r_i}{3}\right)h_i \\ d_i &= y_i \end{aligned}$$

Hasta aquí tenemos explicado el método que utilizaremos mediante una implementación en C++. Ahora es necesario explicar las características del problema particular al cual le aplicaremos el método:

A partir de un archivo de entrada levantaremos secuencialmente pares ordenados, de a uno por vez, supongamos $n+1$ puntos. Si en cada vez que levantamos un punto nuevo consideramos que pasa un instante de tiempo, entonces podríamos escribir nuestra función a estimar de la siguiente manera:

$$f(t) \rightarrow (x, y), f: \mathbb{R} \rightarrow \mathbb{R}^2$$

Donde t representa el instante en segundos y los pares (x, y) representan coordenadas dentro del campo de juego. En particular describen como se desarrolla el desplazamiento de la pelota.

Cada instancia, es decir, cada archivo de los cuales iremos levantando uno a uno los puntos describen (y grafican en el visualizador provisto por la cátedra) funciones de diferentes familias. La imagen de cada una de estas funciones será considerada la trayectoria de cada tiro que nos brinda la instancia analizada. Dicha función quedará en función del tiempo a su vez para que quede bien definida la función, ya que sin la inclusión del mismo como dominio de la función, deberíamos considerar la función f con dominio en una de las coordenadas e imagen de la otra, es decir sería $f(x) \rightarrow y, f: \mathbb{R} \rightarrow \mathbb{R}$. Ahora los pares ordenados que levantamos de los archivos son (x, y) donde x es el elemento del dominio e y es la imagen, es decir $f(x)$. El problema surge, en el escenario donde la instancia presenta al menos dos pares donde la coordenada asociada al dominio (la x) es igual en ambos. En este caso no podríamos aproximar nada, ya que el método que utilizamos está definido para los comportamientos que cumplen las condiciones para ser función. Sobre la línea del arco (la cual se encuentra en $x = 125$) estará ubicado un arquero el cual no puede desplazarse sobre dicho eje, es decir $x = 125$, que es la línea del arco, únicamente podrá moverse sobre el eje y , es decir, sobre la línea del arco, de un lado a otro.

La idea de este arquero es que esté ubicado ahí para atajar las pelotas que van dirigidas al arco (donde atajar, representa, si lo vemos desde el punto de vista gráfico de las funciones que describen las trayectorias de la pelota, a moverse de manera conveniente de manera tal que, en el instante de tiempo que la pelota esté pasando por coordenadas que están dentro del rango de puntos que conforman el arco dentro del área de juego (plano x, y), el arquero se encuentre lo suficientemente cerca y pueda atraparla (luego se entrará en mayor detalle en lo que al arquero respecta). Nuestro objetivo será predecir el movimiento de la pelota a través del campo de juego a partir de los datos que se irán tomando del archivo de entrada (instancia) que, como se dijo antes, nos brindará las coordenadas por las cuales, instante a instante, se vió pasar la pelota (habrán instancias con y sin ruido). A medida que vemos la trayectoria que va describiendo el comportamiento de la pelota sobre el campo de juego, intentaremos aproximar dicho comportamiento mediante una función (en particular usando el método introducido, interpolación polinómica segmentaria mediante Splines Cúbicos), para poder así, una vez estimada la función, estimar el instante de tiempo por el cual la pelota estará pasando por el arco, lo cual eventualmente ocurrirá; pero que, por lo dicho anteriormente respecto a los movimientos del arquero, nosotros utilizaremos esta aproximación considerada una predicción del recorrido final de la pelota para tener un indicio de como indicarle al arquero como desplazarse para que sea capaz de atajarla cuando ese momento ocurra (recordar que el dominio de la función que estimaremos tiene como dominio a los instantes de tiempo en segundos).

Para atacar este problemas, como se dijo anteriormente, aplicaremos el método de interpolación polinómica segmentaria mediante Splines Cúbicos. Para esto, nos resultará conveniente separar la función original a

estimar, $f(t) \rightarrow (x, y), f: \mathbb{R} \rightarrow \mathbb{R}^2$, en dos funciones por separado que serán estimadas por separado, es decir

- $g(t) \rightarrow x, g: \mathbb{R} \rightarrow \mathbb{R}$
- $h(t) \rightarrow y, h: \mathbb{R} \rightarrow \mathbb{R}$

A continuación, aplicaremos el método desarrollado al comienzo de esta sección, de manera independiente, para obtener dos polinomios spline, uno al cual llamaremos S_g y otro S_h , los cuales, respectivamente, interpolan en función del tiempo y nos ayudarán a predecir luego, las coordenadas x e y , por las cuales la pelota pasará instante a instante.

Una vez armados los dos sistemas de los cuales se despejarán las incógnitas que representan las derivadas segunda del spline en cada t_i para $i = 1, \dots, n-1$.

Procederemos a tomar en consideración dichos vectores obtenidos de la resolución de ambos sistemas, y respectivamente hallaremos los coeficientes que componen los splines $S_{n-1,g}$ y $S_{n-1,h}$ (los que corresponden al último subintervalo, para cada variable respectivamente) de la manera que se mostró en el desarrollo del método, es decir los construidos a partir del subintervalo que considera los últimos instantes de tiempo ocurridos.

Nos quedarán entonces:

- $S_{n-1,g} : \mathbb{R} \rightarrow \mathbb{R}$
 $S_{n-1,g}(t) \rightarrow x$
 $S_{n-1,g}(t) = a_{n-1,g} + b_{n-1,g}(t - t_j) + c_{n-1,g}(t - t_j)^2 + d_{n-1,g}(t - t_j)^3$
- $S_{n-1,h} : \mathbb{R} \rightarrow \mathbb{R}$
 $S_{n-1,h}(t) \rightarrow y$
 $S_{n-1,h}(t) = a_{n-1,h} + b_{n-1,h}(t - t_j) + c_{n-1,h}(t - t_j)^2 + d_{n-1,h}(t - t_j)^3$

Si bien estos polinomios conforman parte de las funciones partidas S_g y S_h para $t \in [t_{n-1}, x_n]$, nosotros los utilizaremos extendiendo su dominio, como también su imagen, a los reales, como se puede ver en los items de arriba. En particular, a nosotros nos interesará parte de ese nuevo dominio que establecimos: los $t \in \mathbb{R}$ tal que $t \in [t_{n-1}, \infty)$. La finalidad de esto, es considerar los instantes de tiempo de los cuales aún no se tienen observaciones empíricas. Los mismos brindan la información sobre la función que describe la pelota en su trayectoria en el ultimo tramo observado, pero serán útiles, ya que los mismos nos darán una aproximación de la posible trayectoria que aún no fue observada, en otras palabras, las coordenadas de la pelota en x e y sobre el campo de juego en los instantes posteriores a los que fueron observados. Como se nombró previamente, el arco se encuentra en $x = 125$, también el arquero siempre estará parado sobre esa línea, aunque, lo que sí, puede desplazarse de arriba a abajo sobre el eje y . Esto nos facilita la resolución del problema, ya que, para determinar el principal criterio de decisión del arquero haría falta conocer sobre que valor de y va a pasar la pelota cuando esté sobre $x = 125$, es decir la línea del arco. Así conoceríamos por donde ingresaría la pelota al arco cuando esta llegue a destino.

Es por esto que necesitaremos, por como está definida la aridad de la función, calcular el instante t por el cual la pelota pasa por el $x = 125$ (trabajando sobre el polinomio $S_{n-1,g}$ de la cancha, para luego, conociendo tal instante, determinar por que valor de y (trabajando con el polinomio $S_{n-1,h}$) va a pasar la pelota: armándonos finalmente el par (x,y) que nos dirá, aproximadamente, por donde pasará la pelota cuando llegue el instante en que ingresa en el arco.

Para hallar el instante en el cual la pelota ingresa en el arco, deberemos buscar el cero al spline $S_{n-1,g}$ (que está asociado a la coordenada x de la pelota en su trayectoria instante a instante). Como se dijo, luego, se evaluará dicho instante en $S_{n-1,h}$ (asociado a la coordenada y de la posición de la pelota) y obtendremos la estimación deseada. Utilizaremos un método, ya mencionado en la introducción de este informe:

Método de la bisección: Mediante este método calcularemos el cero de la función $S_{n-1,g}$ aunque, como dijimos, debemos realizar una pequeña modificación para utilizarlo en el contexto de nuestro problema a resolver. Como a nosotros no nos interesa el t (llamémoslo t_r) tal que $S_{n-1,g}(t_r) = 0$, que es lo que estaríamos calculando mediante este método que justamente aplica para el cálculo de los ceros/raíces de las funciones, sino el t_r tal que $S_{n-1,g}(t_r) = 125$, lo que haremos será aplicar el método a la siguiente función:

$$P(t) = S_{n-1,g}(t) - 125$$

Entonces ahora, encontrarle el cero a esta función equivaldría a encontrar el t_r tal que $S_{n-1,g}(t_r) = 125$ ya que, suponiendo ese t_r , valdría que:

$$P(t_r) = 0 = S_{n-1,g}(t_r) - 125 \text{ entonces } S_{n-1,g}(t_r) = 125$$

El pseudocódigo del método es el siguiente:

```

bisección(a ∈ ℝ , b ∈ ℝ , P : ℝ → ℝ )
for i = 1, . . . , N
    x = (a+b)/2.0
    if (abs(P(x)) ≤ EPSILON)
        return x
    else
        if (P(x) P(a) > 0)
            a = x
        else
            b = x

```

Donde la función $\text{abs}(a)$ devuelve el valor absoluto de a .

Aclaraciones importantes para asegurar la convergencia del método: primero, es necesario encontrar y tomar a y b de manera tal que $P(a)P(b) < 0$. Es fundamental conocer además un poco la función a la cual se le calculará el cero, ya que, como precondition la misma debe ser continua en el intervalo $[a, b]$. Lo que se hace con este método es, lo que se podría llamar, una búsqueda binaria de la raíz ya que uno comienza tomando dos puntos (a y b) donde la función difiere en signo cuando se evalúa dichos dos valores por separado, y, por la continuidad de la misma, existe seguro al menos un valor para el cual la función cruza el eje de coordenadas, es decir, se vuelve 0.

En cada iteración se partía el intervalo creado por a y por b a la mitad y según como sea el signo de dicho valor medio (x en el pseudocódigo) evaluado en la función volveremos a iterar el método pero ahora tomando la mitad del intervalo

determinado por a y b , en particular nos quedaremos con el intervalo que aún presente cambio de signo, si no se presenta es o bien porque encontramos la raíz, o porque no estamos en la mitad del intervalo que contiene a la raíz buscada entonces descartaríamos esa mitad y continuaríamos iterando en la restante (en cada iteración tomaremos $[a, x]$, asignando $b = x$ o bien $[x, b]$, asignando $a = x$ dependiendo el caso en el que se caiga de los nombrados).

Otra cosa importante es el criterio de parada del algoritmo, como se puede ver en el pseudocódigo la parada del algoritmo esta dada, o bien por encontrar un valor de la función tal que, en módulo, sea lo suficientemente chico (menor o igual que el epsilon, que es un valor más chico que 1, por ejemplo 0.00001) y ahí considerar que ya estamos en un valor el cual es raíz de la función a la cual se le aplicó el método, o bien por el término de las iteraciones dadas por el Ciclo For impuesto en el comienzo del algoritmo. La idea es poner un N lo suficientemente grande, este criterio se lo considera pesimista pero en la práctica, y conociendo levemente la función, funciona correctamente, ya que, usualmente se encuentra dicha evaluación con el x que será considerado raíz que da menor que el Epsilon impuesto.

Una vez encontrado dicho t tal que $S_{n-1,g}(t) = 125$ restaría evaluar $S_{n-1,h}(t)$ y así, obtener en el eje y por donde pasará la pelota cuando esté sobre la línea del arco ($x = 125$). Esta será, como se dijo anteriormente, nuestra forma de predecir instante a instante sobre donde estará la pelota aproximadamente cuando llegue al arco. Con estas últimas aclaraciones damos por terminada el desarrollo teórico que explica nuestra implementación y como fue el procedimiento seguido para enfrentar el problema planteado.

El criterio del arquero: Una parte muy importante en el desarrollo del TP fueron las decisiones que tomamos con respecto a los movimientos a realizar por el arquero en función de los datos que obteníamos para cada instante del tiro sobre la supuesta posición final de la pelota. Luego, pasamos a enumerar los parámetros que tuvimos en cuenta para intentar darle al arquero la mayor intuición de arquero posible. Más adelante, en experimentación y discusión justificaremos la razón de las mismas (en particular de los números totalmente arbitrarios que elegimos para las comparaciones).

1) La primera decisión importante fue en cuanto a las jugadas en las cuáles la pelota pega en un jugador: en esos casos, consideramos que al pegar en un jugador, la pelota podría cambiar abruptamente su trayectoria. Con lo cual, lo más lógico sería, en ese momento, borrar toda la información del tiro que teníamos hasta ese momento y calcular la nueva trayectoria de la pelota sólo a partir de los puntos recorridos por la pelota a partir de ese instante.

2) La segunda decisión fue la realización de dos tipos de movimientos para el arquero: a) que el arquero se mueva lo máximo posible hacia la supuesta dirección final de la pelota y b) que el arquero se mueva lo máximo posible hacia el centro del arco. Entonces, en función de varios parámetros vamos a mover al arquero de la forma a) o b).

Pasamos a enumerar dichos parámetros: i) Si la diferencia entre la última medición de posición final de la pelota y la anteúltima es muy grande (que sea mayor a 100 -donde el tamaño del arco es de 70-) ó si la última posición final de la pelota calculada esta fuera del arco, entonces separamos en dos casos: si la cantidad de puntos considerada hasta el momento (T) es pequeña (menor a 5), dejamos al arquero quieto; si el movimiento anterior del arquero no fue nulo y $T < 5$, entonces que se mueva de la misma forma que en el turno anterior (porque probablemente el arquero se venía moviendo bien y esta última medición tuvo error); sino, se mueve hacia el centro del arco (el arquero no se movió en el turno anterior, y no hay manera de saber si el último tiro tuvo mucho error o no).

Por otro lado, si la diferencia entre la última y anteúltima medición no es tan grande pero tampoco tan pequeña (entre 20 y 40), movemos al arquero lo máximo posible hasta donde la última medición lo indica (porque asumimos que el cálculo de la posición final de la pelota esta convergiendo).

3) La tercera decisión importante que influyó mucho en la calidad de atajada de nuestro arquero fue la implementación de un método para decidir cuándo considerar los cálculos de posición final de la pelota hecho por el método Splines y cuándo no. Esto es muy importante ya que, como se verá en la sección Resultados, dicho método es bastante propenso a devolver cada cierta cantidad de valores alguno que no se condice correctamente con el resto. Entonces lo que hicimos fue decidir que sólo consideraríamos como válidos los valores devueltos que estuvieran precedidos por 3 valores devueltos bastante similares entre ellos y bastante similares a este último (que la norma dos entre los puntos sea menor a 10). De lo contrario, consideramos que la función Splines nos devolvió el mismo valor que en el último llamado que habíamos hecho y en función a ello movemos al arquero.

4) Otra decisión que supusimos que mejoraría la atajada del arquero fue considerar como parte del tiro sólo a los puntos que aún no hayan pasado por $x=125.0$ Es decir, cuando la pelota aún no salió de la cancha.

Como compilar los archivos fuente y ejecutar tests: Para compilar los archivos fuente del TP, en Linux (Ubuntu), es necesario abrir una terminal ($\text{ctrl} + \text{alt} + \text{t}$), acceder a la carpeta donde se encuentran los archivos fuente `main.cpp` (mediante el comando `'cd 'ruta de la carpeta''`).

Luego, teclear el siguiente comando en la consola:

```
g++ main.cpp -std=c++11 -o main
```

(Es necesario tener instalado una versión de gcc posterior a la 4.7 para utilizar el flag `'-std=c++11'`, requerido para compilar nuestra implementación)

Una vez hecho esto, nos aparecerá en la misma carpeta donde se encuentra `main.cpp`, un archivo con nombre `'main'`, el cual ejecutaremos desde la línea de comandos para realizar los tests, a partir de las instancias de tiro provistas, de la

siguiente manera (este programa toma 3 parametros, los cuales se separan por un espacio cuando se los escribe en la linea de comandos): el primero es un archivo con informarci3n sobre los par3metros de entrada asociados a la posici3n inicial del arquero, su capacidad de movimiento, las dimensiones del arco y por 3ltimo, los puntos que iremos levantando secuencialmente que representan la posici3n actual de la pelota instante a instante. El segundo proveer3 la informaci3n sobre las posiciones de los jugadores que intervienen (o no, puede ser vacio o no tocar la pelota) en la trayectoria del tiro generado por la instancia, y el tercer par3metro es el archivo donde se guardar3n, en base a nuestros criterios de decisi3n, los desplazamientos realizados por el arquero en cada instante:

Ejemplo de uso del archivo main con sus tres parametros:

```
./main visualizador/instancias/cuadraticas/c1.tiro visualizador/jugadores/vacio.jug resultado.arq
```

Una vez generado el archivo resultado.arq (se le puede poner un nombre arbitrario, pero hablaremos de este mismo para continuar con el ejemplo de uso nombrado), podremos visualizar si, efectivamente, el arquero logra moverse lo suficiente seg3n nuestros criterios, a donde supusimos que ir3a la pelota seg3n la trayectoria descrita por la misma, para atajarla y evitar el gol. Para esto, ser3 necesario posicionarse desde consola mediante el comando 'cd' en la carpeta "visualizador" (tippear 'cd visualizador') que se encuentra dentro de la misma carpeta donde se hallaba el archivo main que compilamos y ejecutamos en primer lugar. Una vez hecho esto, correremos el visualizador provisto por la c3tedra que toma 3 par3metros por l3nea de comandos, el primero ser3 el archivo que generamos con el ejecutable main, es decir 'resultado.arq', el que proporciona los movimientos instante a instante del arquero; el segundo, ser3 el mismo archivo que utilizamos como segundo parametro a la hora de correr el main, es decir, el que provee la informaci3n sobre la posici3n inicial del arquero, y provee los puntos que fueron levantados instante a instante; en este caso ser3n utilizados por el visualizador para graficar la pelota moviendose a trav3s del campo de juego describiendo la funci3n que intentamos estimar mediante nuestro m3todo, y, finalmente, el tercer par3metro para el visualizador ser3 la posici3n de los jugadores para poder graficarlos en el 3rea de juego. Un ejemplo de como correr este visualizador es el siguiente (siguiendo con el ejemplo anterior):

```
python main.py ../resultado.arq visualizador/instancias/cuadraticas/c1.tiro jugadores/vacio.jug
```

Todo esto ser3 suficiente para poder correr y comprobar la eficacia de nuestra implementaci3n. A continuaci3n pasaremos a la secci3n dedicada al planteo de las hip3tesis para luego experimentar, discutir y sacar conclusiones del m3todo planteado.

- Hip3tesis:

Para la realizaci3n del m3todo implementado en el TP (Splines c3bicos) realizamos varios experimentos. Pensamos que si el m3todo Splines C3bicos tuviera alg3n sentido y sirviera para resolver el problema que se nos plantea entonces hay algunos experimentos que pueden servir para verificar esto. Aparte de estos experimentos, una vez que hubimos corroborado que el m3todo de interpolaci3n mediante Splines C3bicos serv3a para resolver el problema de ser capaces de intuir la posici3n futura de la pelota a partir de las posiciones anteriores de la misma, tambi3n tuvimos que realizar otros, necesarios para analizar si, efectivamente, el arquero atajaba los tiros en distintos tipos de instancia.

Hip3tesis del experimento 1 :

En este experimento pretendemos analizar si el c3lculo de la posici3n de la pelota al atravesar el arco converge hacia el valor correcto. Tambi3n analizaremos cu3ntos puntos necesita considerar para que la aproximaci3n sea buena. Esperamos no necesitar absolutamente todos los puntos de la trayectoria de la pelota para obtener una aproximaci3n buena para que, as3, el arquero se comience a mover hacia la posici3n final de la pelota a tiempo (es obvio que sino, el arquero no ser3a capaz de atajar la pelota, por m3s de que el m3todo haya calculado correctamente la posici3n final del bal3n).

Hip3tesis del experimento 2 :

En este experimento, pretendemos analizar m3s profundamente el m3todo de interpolaci3n mediante Splines C3bicos, analizando no s3lo la convergencia de la soluci3n final a nuestro problema (la convergencia hacia la correcta posici3n final de la pelota cuando $x=125$ -es decir cuando la pelota atraviesa el arco-), sino tambi3n qu3 sucede con la funci3n que devuelve la interpolaci3n. Ser3a interesante analizar que la interpolaci3n devuelva una funci3n que no s3lo converga en el punto que estamos analizando ($x = 125$) sino que tambien vaya convergiendo progresivamente (cuanto mayor sea la cantidad de puntos de la trayectoria de la pelota considerados) hacia la curva de una determinada funci3n (que deber3a tener forma recta, cuadr3tica o c3bica al realizar el gr3fico de $(x(t), y(t))$ para

algún subconjunto de valores de 't' determinados por nosotros).

Hipótesis del experimento 3 :

En este experimento, analizamos la imprecisión en el cálculo final de la posición de la pelota. Es decir, la presencia de casos en los cuales para intervalos de tiempos de lectura de la trayectoria de la pelota, la interpolación realizada devuelve una función que, para el t tal que $x=125$, la posición en el eje 'y' es muy distinta a la calculada en los instantes anteriores. Es decir, casos en los que el último punto considerado hasta ese entonces haga que la predicción indique incluso que la pelota ni siquiera va a entrar al arco al momento de llegar a la línea del mismo. Esperamos ver que, si hubieran casos como estos solo ocurran en instantes de tiempo donde aún la pelota se encuentra bastante lejos del arco y no se tiene una clara noción de la función asociada que describe la trayectoria de la pelota o bien porque el disparo es ruidoso, ya que la trayectoria de la pelota será de cambios bruscos e impredecibles. Para solucionar estas situaciones podríamos hacer que el arquero se mueva acorde a los puntos anteriores que, al menos, nos brindaron resultados confiables, donde con confiables nos referimos a que por lo menos la predicción nos dice que la pelota se dirige hacia el arco y no a cualquier otra parte, y no al último, que se supone generó un error en nuestros cálculos. Si esto no fuera así, entonces claramente nuestro método no sería útil pues sería más difícil identificar estas situaciones, con lo cual al entrar en ellas, por ejemplo, el arquero comenzaría a moverse en esas direcciones de forma absurda (sería equivalente a intentar atraparla afuera del área del arco cuando ya sabíamos que siempre ingresan en el mismo) y, cuando finalmente la pelota se aproxima al arco (como precondition en los disparos, sabemos y asumimos que todos, al final, ingresan en el mismo), y el próximo spline calculado reconozca dicha trayectoria final hasta su ingreso, el arquero quizá se encuentre bastante lejos y no le de el tiempo suficiente para desplazarse de regreso al sitio donde realmente pasará la pelota.

Hipótesis del experimento 4 :

También, creemos que sería interesante analizar la calidad en la atajada de nuestro arquero. Es decir, una vez realizados los tres experimentos anteriores, que nos confirmaron que nuestra forma de afrontar el problema con el que nos enfrentamos tiene algún sentido, verificar si realmente somos capaces de atajar la pelota. Para esto, analizamos varias instancias, y calculamos el porcentaje de atajadas del arquero, modificando los criterios del arquero para atajar la máxima cantidad de tiros posibles. Es decir, analizar qué sucede cuando modificamos los criterios de nuestro arquero (los criterios que ya explicamos en la sección Desarrollo).

3. Resultados

Resultados del experimento 1 :

Para estos experimentos se utilizaron los archivos r2.tiro, cbp4.tiro y c3.tiro respectivamente. Con respecto al gráfico, los T representan a la cantidad de puntos que se tomaron para calcular cada spline, mientras que t son los puntos en los que fue evaluado la función.

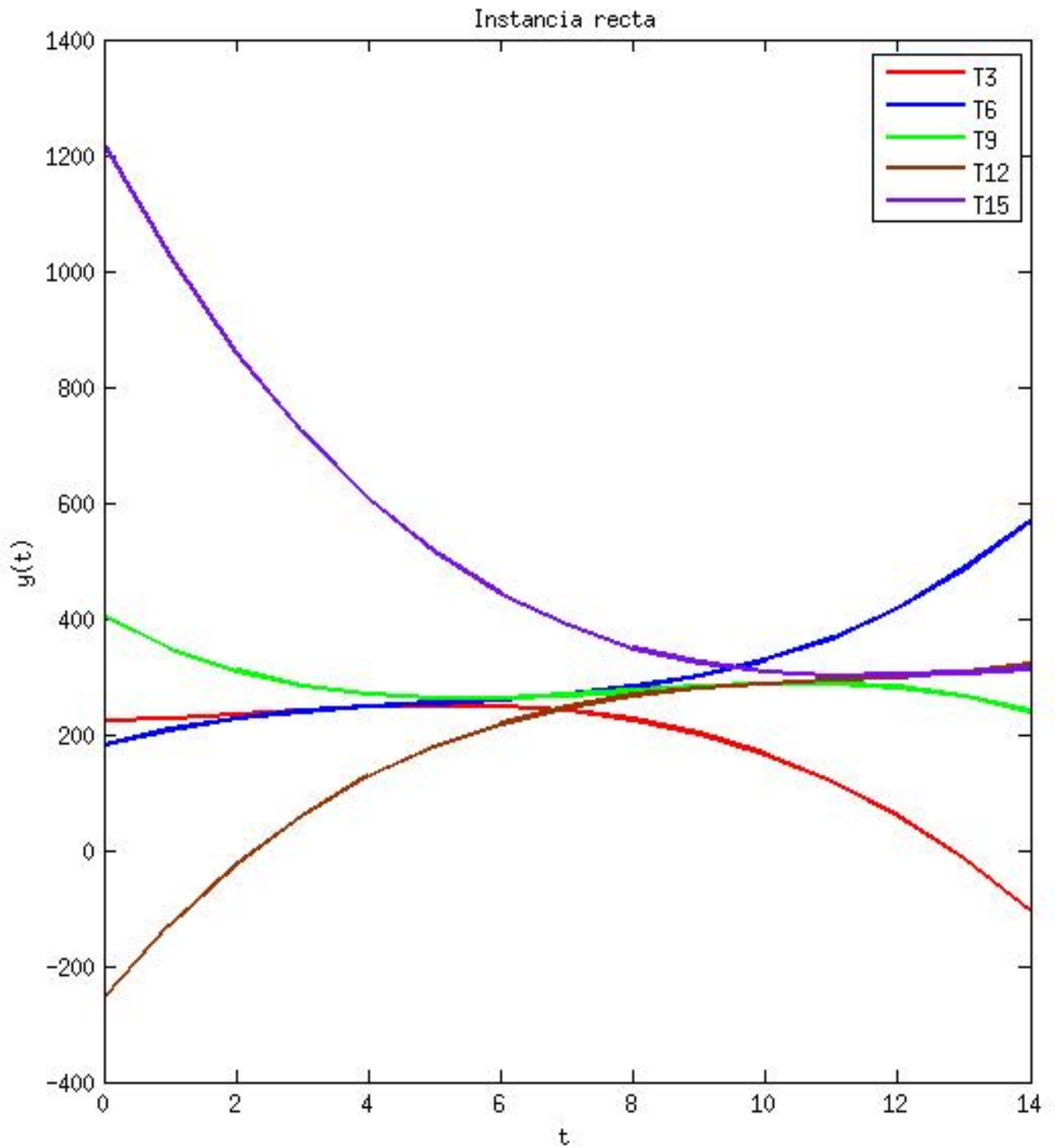


Fig 1.1

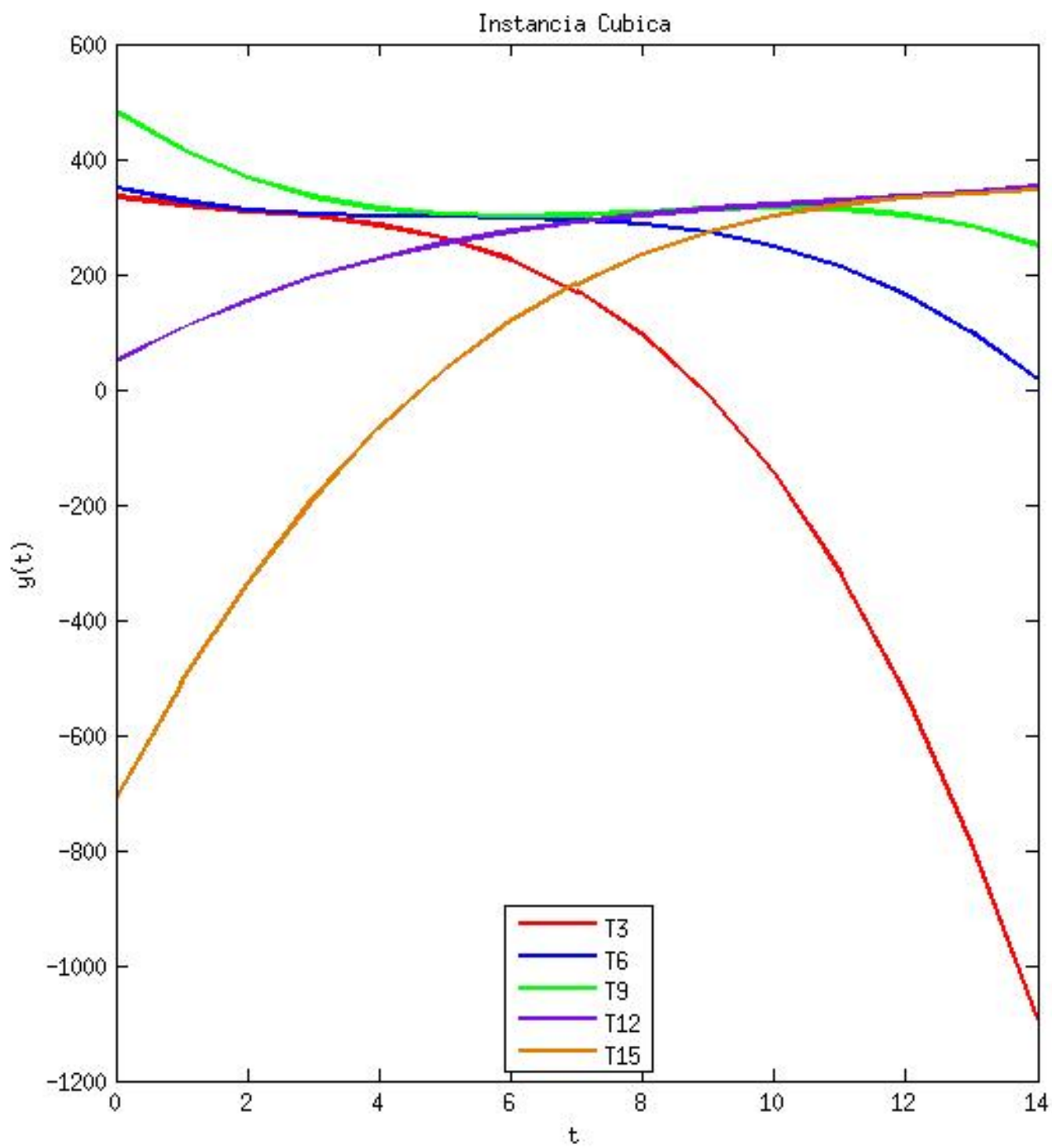


Fig 1.2

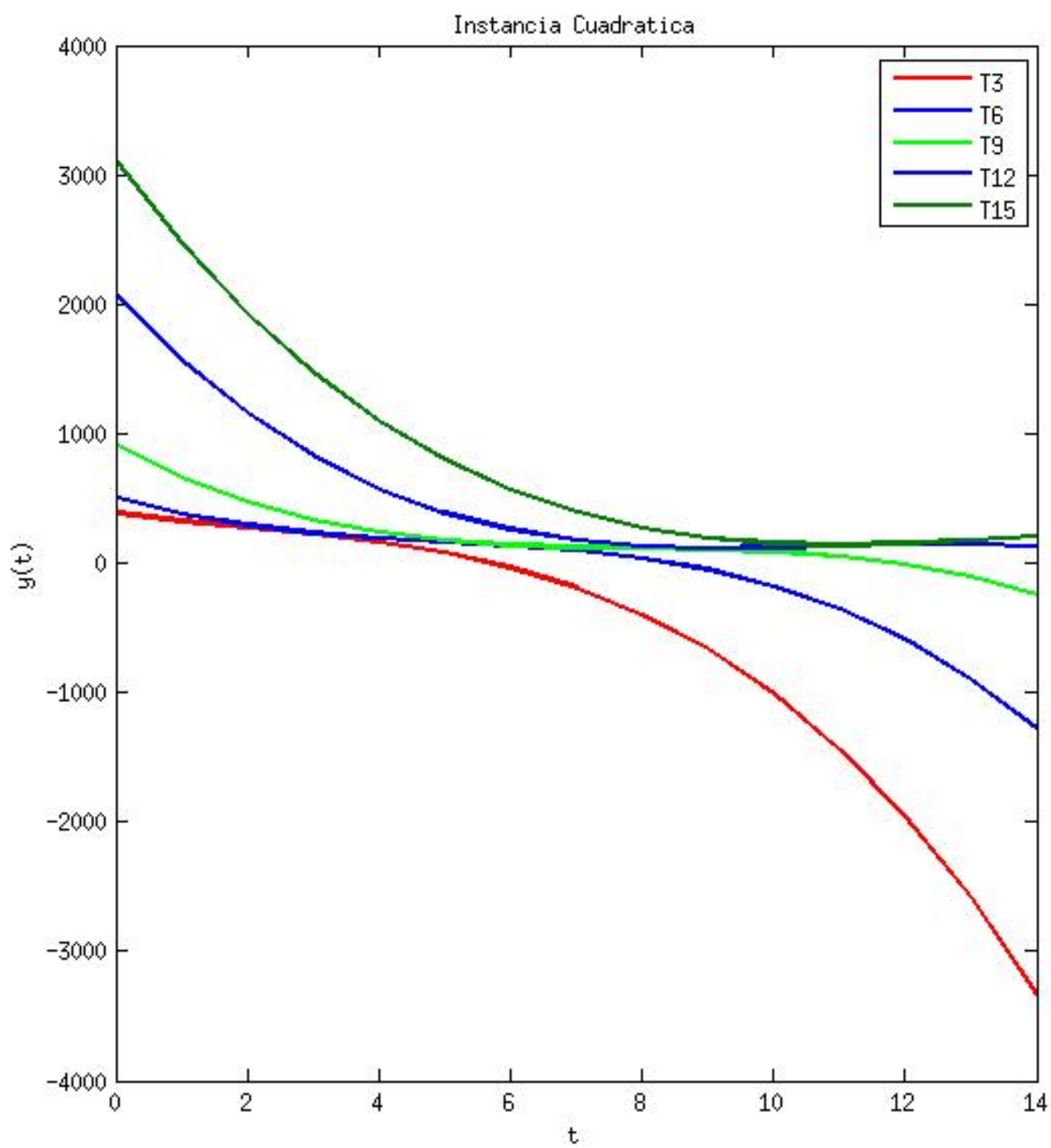


Fig 1.3

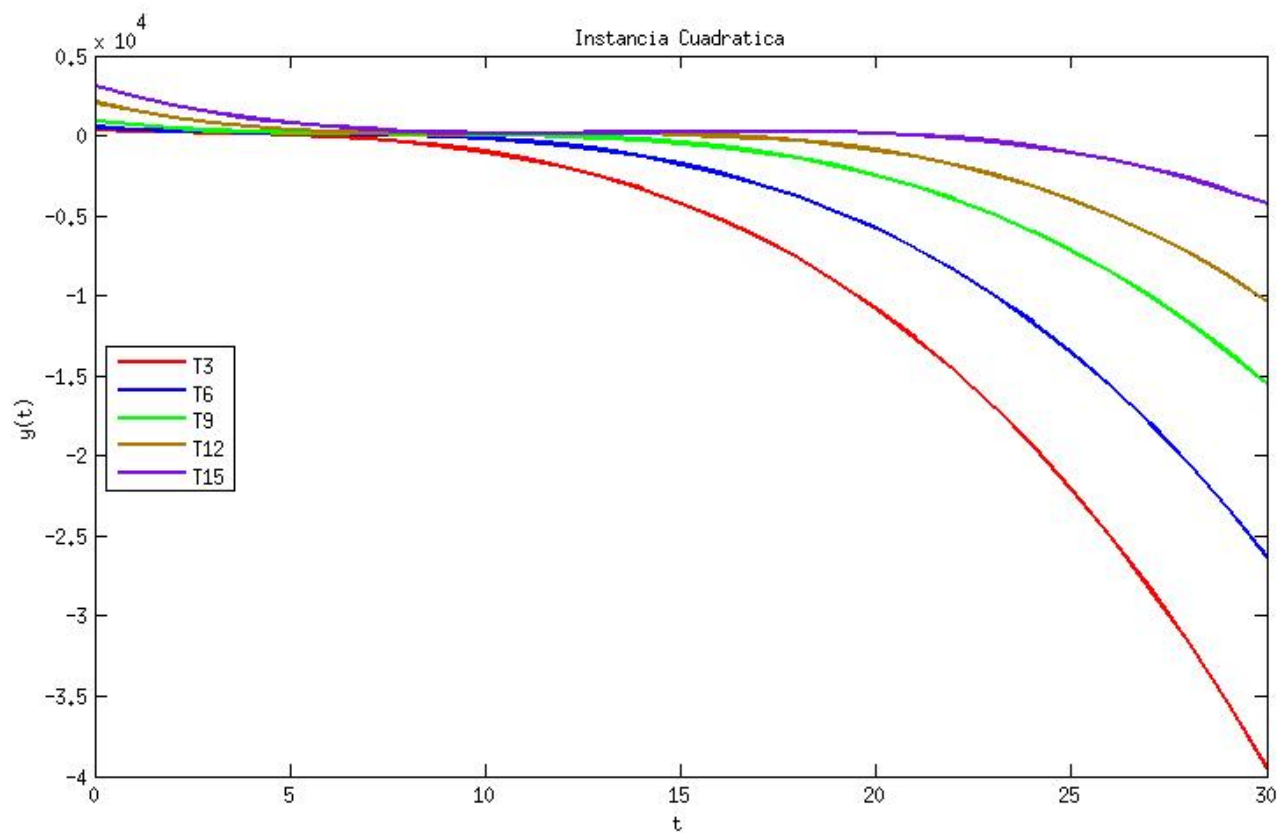


Fig 1.4

Resultados del experimento 2 :

Estos gráficos arrancan en $x=125$, se utilizaron los archivos r2.tiro, cbp4.tiro y c3.tiro respectivamente. Con respecto al gráfico, los T representan a la cantidad de puntos que se tomaron para calcular cada spline, mientras que t son los puntos en los que fue evaluado la función

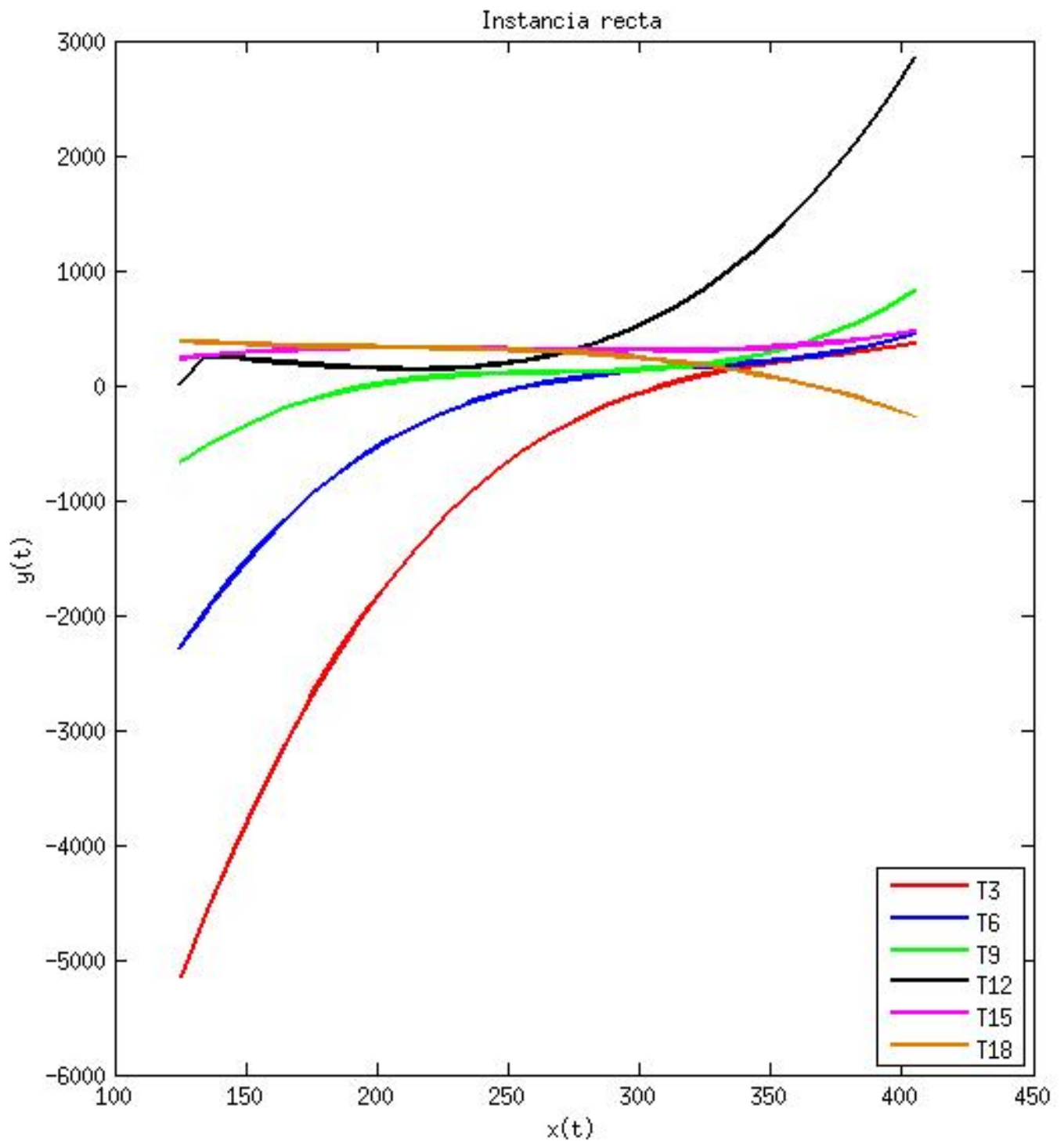


Fig 2.1

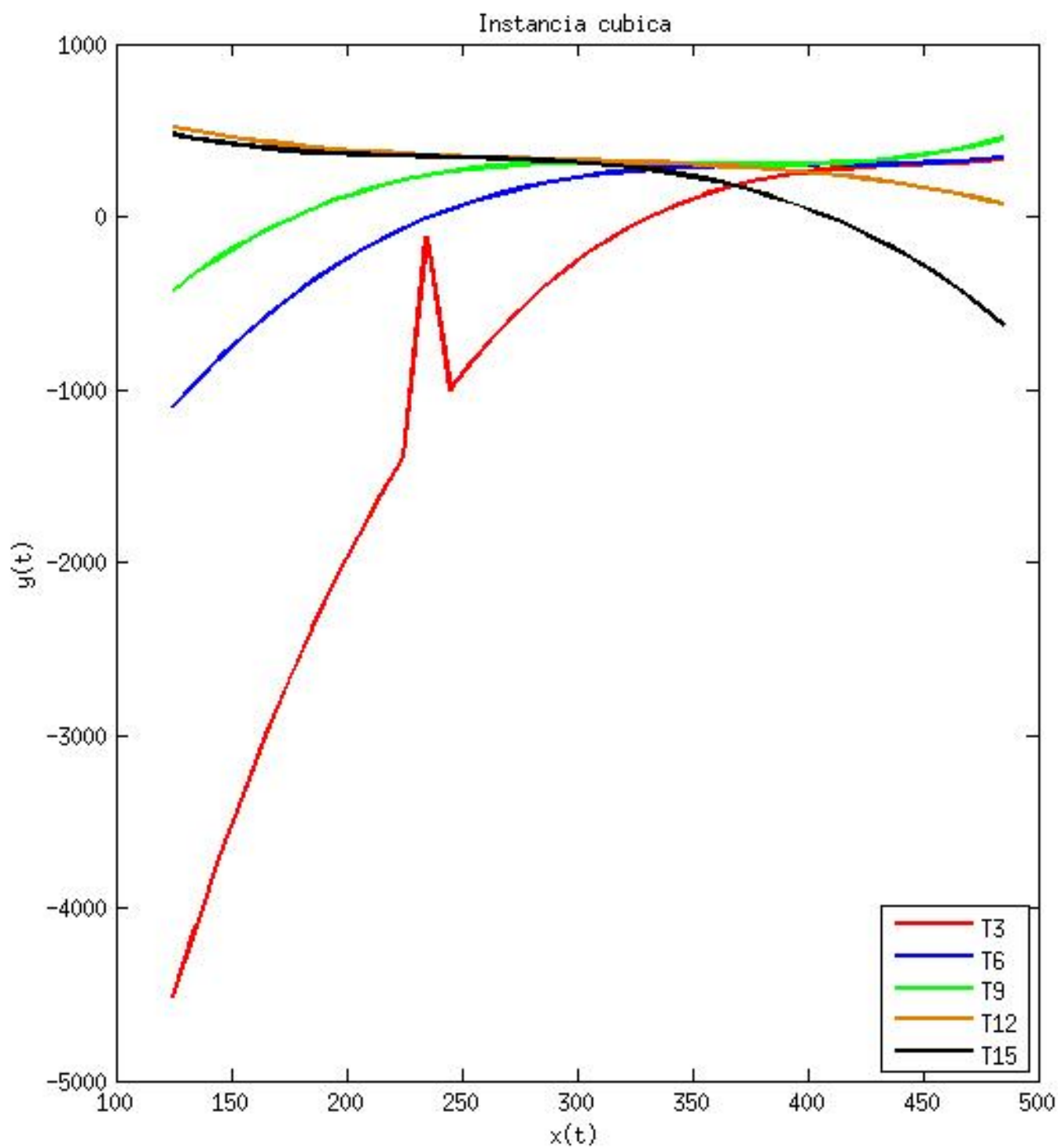


Fig 2.2

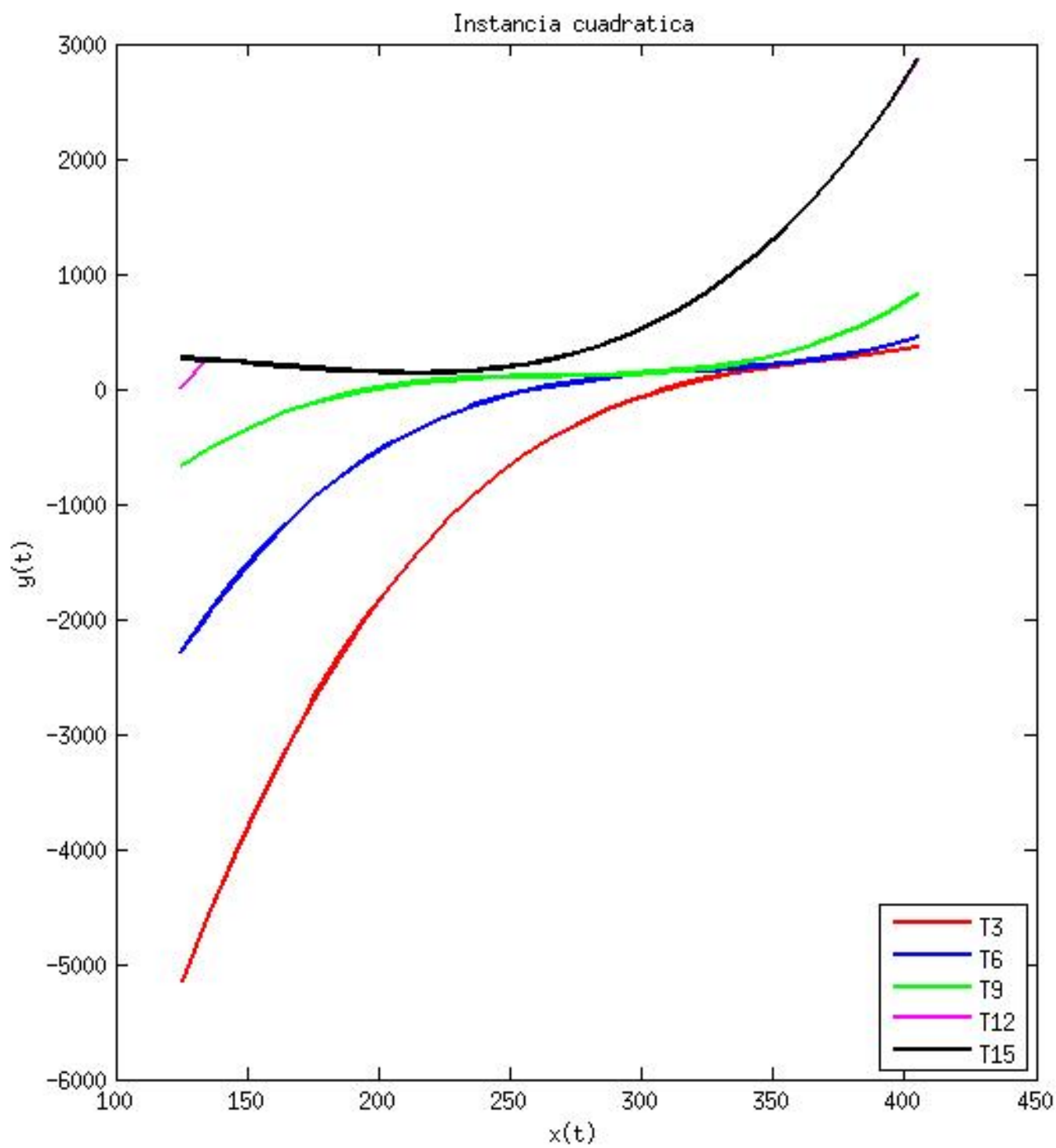


Fig 2.3

Resultados del experimento 3 : En este experimento se analizará, instante a instante (es decir a medida que va creciendo T , que también representa la cantidad de puntos observados empíricamente), cual es la posición final de la pelota en el arco en el eje de coordenadas y (es decir, cuando $x = 125$ para el t que nos brinda el último polinomio calculado mediante el método de la bisección, el mismo es el tal que evaluado en $S_{n-1,g}(t) = 125$, esto queda implícito).

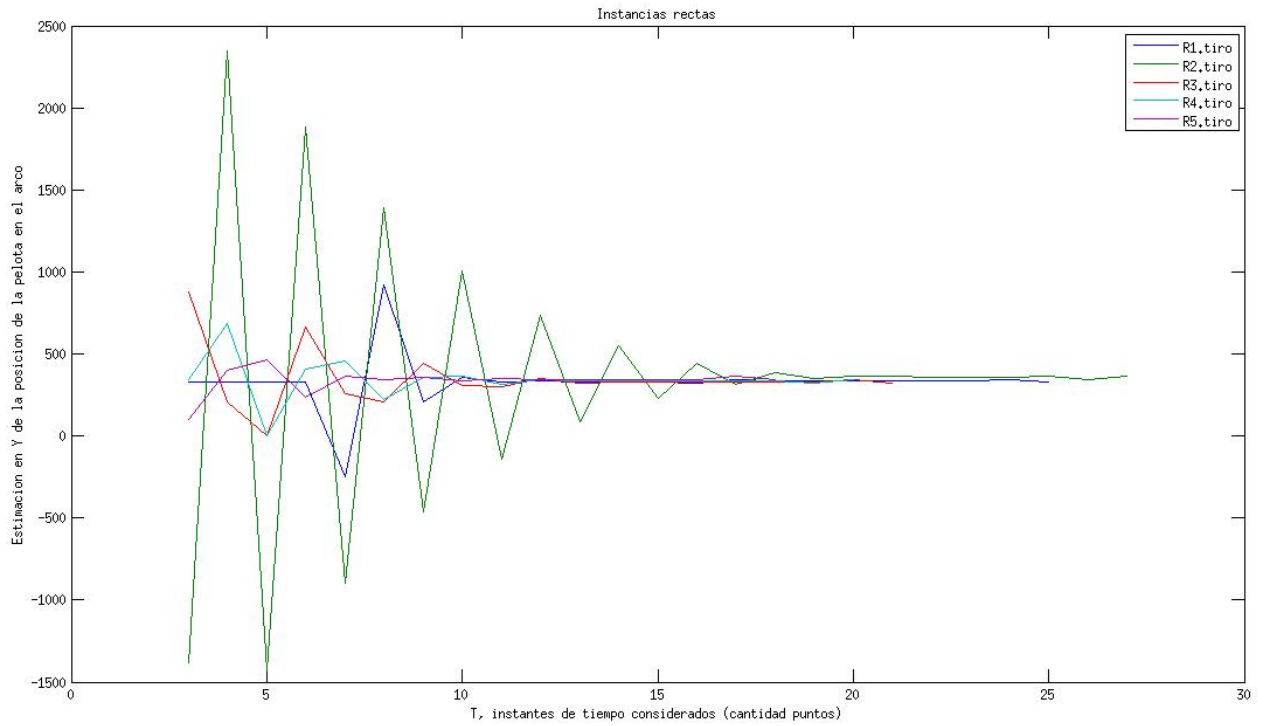


Fig 3.1

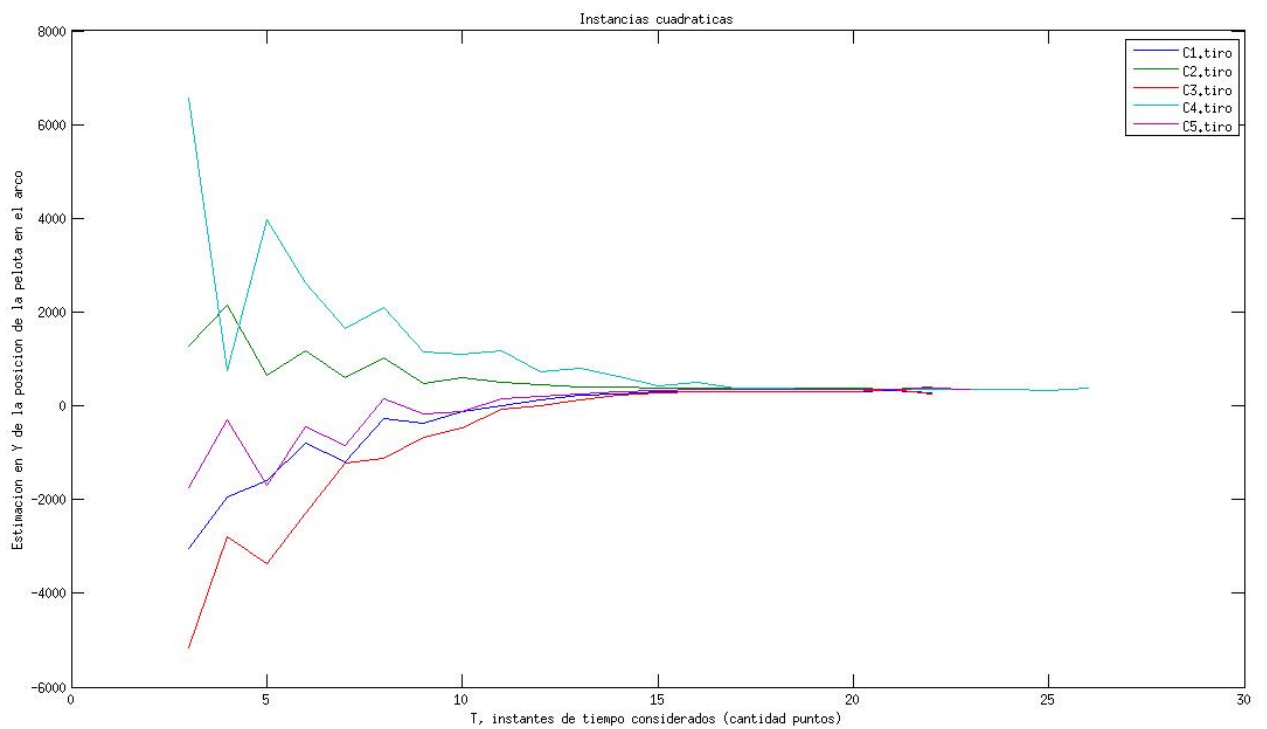


Fig 3.2

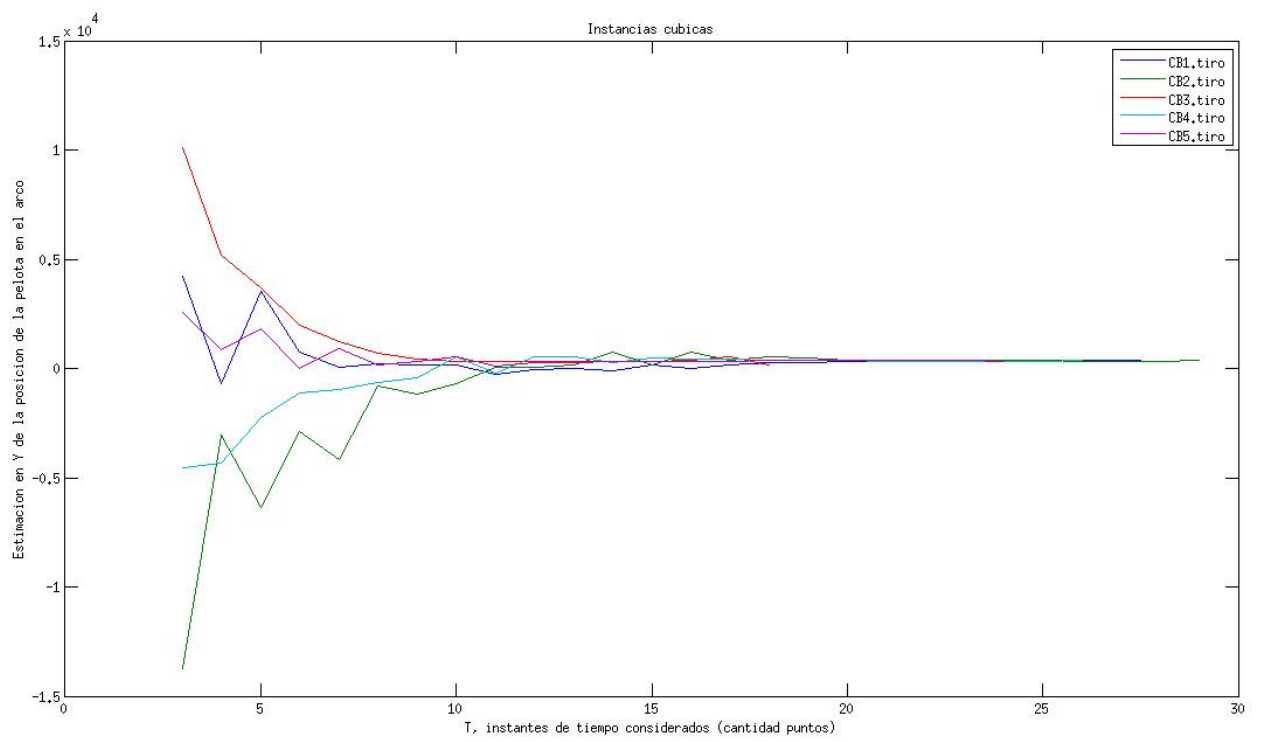


Fig 3.3

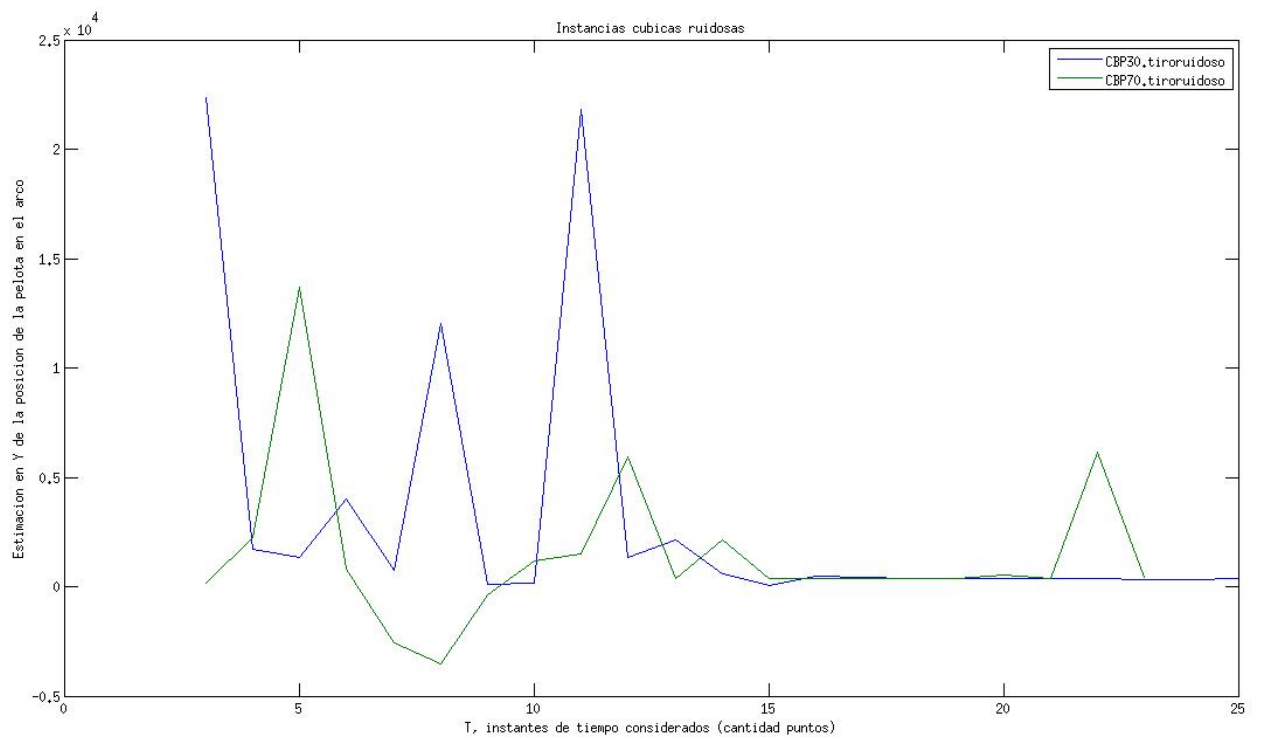


Fig 3.4

Resultados del experimento 4 :

Las O significan que el arquero atajo, mientras que las X significa que no. Los tiros fancy no atajo ni uno. los numeros de 1 a 5 son los archivos que use en cada experimento, es decir del tiro 1 al 5. Sin embargo, modificando algunos de los parámetros que tomamos para efectuar los movimientos del arquero sí logramos en atajar el tiro libre en un momento. Pensamos que esto se debió a que hicimos una buena combinación de criterios de atajada del arquero y de parámetros considerados.

i) Sacando los criterios: todos

	1	2	3	4	5	Tasa de aciertos
recta	O	O	X	X	O	3/5
cúbica	X	O	X	O	X	2/5
cuadrática	X	X	X	X	X	0/5

ii) Sacando los criterios: en vez de $x > 0$, consideramos $\text{solox} > 125$.

	1	2	3	4	5	Tasa de aciertos
recta	O	O	X	X	O	3/5
cúbica	O	O	X	O	O	4/5
cuadrática	X	X	X	O	O	2/5

iii) Sacando los criterios: si splines calcula medianamente aceptable

	1	2	3	4	5	Tasa de aciertos
recta	O	O	X	O	O	4/5
cúbica	O	O	X	X	X	2/5
cuadrática	X	X	X	O	X	1/5

iv) Sacando los criterios : si splines calcula cualquier cosa:

	1	2	3	4	5	Tasa de aciertos
recta	O	X	X	X	O	2/5
cúbica	O	O	X	O	O	4/5
cuadrática	X	O	X	X	X	1/5

v) Sacando los criterios: no considero las mediciones que difieren mucho de las ultimas 3:

	1	2	3	4	5	Tasa de aciertos
recta	O	O	X	X	O	3/5
cúbica	O	O	X	O	O	4/5
cuadrática	X	X	X	O	O	2/5

vi) Sacando los criterios: ninguno

	1	2	3	4	5	Tasa de aciertos
recta	O	O	X	O	O	4/5
cúbica	O	O	X	O	O	4/5
cuadrática	X	X	X	O	O	2/5

Atajadas del arquero en tiros con Ruido (instancias2) probados con los 6 criterios nombrados arriba (desde el 1 hasta el 6), son 10 instancias de tiros ruidosos:

El arquero atajó:

i) 6/10 sacando todos los criterios

ii) 6/10 sacando el criterio de tomar $x > 125$ en vez de $x > 0$

iii) 6/10 sacando el criterio en de que splines calcula medianamente aceptable

iv) 5/10 sacando el criterio de que splines calcula cualquier cosa

v) 6/10 sacando el criterio de no considerar las mediciones que difieren mucho de las ultimas 3

vi) 6/10 sin sacar ningun criterio

4. Discusión

Anomalía observada en Fig 1.1 y 2.1 Dicha anomalía suponemos que se debe a un error de graficación nuestro, pues se supone que la trayectoria de los tiros que usamos para hacer los tests son cuadráticas, rectas o cúbicas. Luego, en principio no tendría sentido que la función que devuelve nuestra interpolación, que sabemos que en este caso es una función cúbica, presente dicha anomalía. Este tipo de anomalías si son esperables, por el contrario, en los otros gráficos, de $(t, y(t))$.

Sobre el experimento 1 Sobre el gráfico de $(t, y(t))$ en el caso en que el recorrido de la pelota es cuadrático, se observa que a medida que se considera una mayor cantidad de puntos de la trayectoria de la pelota, para cualquier t en particular que consideremos (especialmente mirando a partir de $t=8$), las curvas de cada una de las funciones que devuelve la interpolación se acercan progresivamente hacia un mismo valor sobre el eje $y(t)$ -al menos aproximadamente-. Entonces, esto nos da la pauta que si ese valor fuera la posición final real de la pelota, entonces estamos bien encaminados (eso lo analizaremos en el experimento 2). Además, en el caso particular considerado para la realización del gráfico, se observa que tan sólo considerando $T=9$ posiciones de la pelota (es decir, menos de la mitad de las posiciones que va a recorrer la pelota antes de llegar al arco), la curva se toca con la descrita por la curva realizada con $T=15$ precisamente en el momento en que $y(t)$ toma el valor que nosotros buscamos. Es decir, que tan sólo considerando la mitad de los puntos de la trayectoria de la pelota, ya fuimos capaces de obtener la ecuación de una curva muy similar a la que se podría obtener con 15 o más puntos. Suponemos que eso debería permitir al arquero comenzar a moverse a tiempo hacia el lugar correcto para llegar a atajar la pelota.

En el caso de la instancia cúbica se observa algo similar: en este caso, las funciones que obtenemos con valores de T cada vez mayores tienen la propiedad de que para cada t que consideremos, el valor de $y(t)$ se va acercando cada vez más hacia el esperado, hasta que para $T=12$ y $T=15$, las funciones dan prácticamente lo mismo para una cantidad considerable de puntos de las mismas.

Algo a destacar, es que todas estas funciones toman valores muy distintos si las comparamos en valores de t alejados del t que hace que $x(t)=125$. Para analizar esta problemática es conveniente analizar el segundo gráfico de la instancia cuadrática considerada para decir la razón por la cuál pensamos que esto no nos afecta a los efectos del TP: allí se pueden observar dos cosas: la primera, que al incrementar el valor de T , cada curva se va acercando progresivamente más a la construida con el T mayor, y cada una de estas curvas se aplanan más a partir de un punto. La segunda, a medida que se considera un T mayor, el intervalo de valores de t para los cuales la curva de la función se hace recta, tomando valores cercanos al $y(t)$ final esperado antes de que la curva se haga convexa es mayor. Esto nos dice que al considerar una mayor cantidad de puntos de la trayectoria del tiro, la precisión con la que calculamos la posición final de la pelota es mayor. Es decir, el cálculo de la posición final de la pelota converge. Eso es bueno.

En el caso de un tiro recto, sucede lo mismo que en los tiros cuadráticos y cúbicos, con la diferencia que las curvas no se acercan progresivamente a la que se realiza con el T mayor sino que vamos obteniendo siempre curvas con valores muy distintos, pero que siempre se van acercando, por arriba o por abajo, a un valor de $y(t)$: a partir de un valor de T se aplanan y toman el mismo valor de $y(t)$ para un intervalo grande de valores de t .

Es interesante notar que en los tres tipos de tiros, las curvas de los gráficos de $(t, y(t))$ se comienzan a pegar aproximadamente para el mismo valor de T . Eso quiere decir que la calidad con la cuál calculamos la posición final de la pelota es similar en cada una de las instancias que consideramos.

Sobre el Experimento 2 :

En los gráficos correspondientes a este experimento de $x(t) \rightarrow y(t)$ se puede observar que tanto para instancias rectas, cúbicas y cuadráticas, cuanto mayor es el T que consideramos, la curva de la función que va de $x(t) \rightarrow y(t)$ se asemeja progresivamente más a la forma característica de una recta (en el caso de rectas), a la de una función cuadrática (en el caso de la cuadrática), y a una cúbica (en el caso de la cúbica). Es interesante observar que se puede decir que en los tres casos, considerando $T=15$ la curva que graficamos ya adopta la forma que esperamos en cada caso. Esto quiere decir que el método de interpolación mediante Splines Cúbicos sirve para aproximar correctamente los tres tipos de

funciones. Especialmente si se observa que en $x=125$, efectivamente la función toma el valor de $y(t)$ esperado, según el tiro que fue efectuado.

En el caso de la instancia recta vale observar que considerando $T=18$, la curva que graficamos es aproximadamente una recta para un intervalo importante de valores, pero luego se curva, como si fuera una cuadrática. Esto se podría decir que es una desventaja de nuestro modelo, ya que al usar siempre splines cúbicos, en el caso de interpolar rectas, el modelo siempre va a tender hacia curvas cúbicas. Es interesante notar que, por el contrario, esto no sucede en la instancia cuadrática (para valores de T menores que 12, se interpola hacia una función cúbica pero ya cuando tenemos 12 puntos a interpolar (la mitad de la totalidad de los puntos que tenemos en el .tiro), la función graficada tiene la forma de una cuadrática.

Sobre el Experimento 3 : Como se planteó en la hipótesis, en este experimento discutiremos la presencia de malas predicciones en base a la cantidad de puntos (instantes de tiempo observados) que se considere según instancia de tiro (familia de función) que se esté probando para aplicar el método de Splines Cúbicos. En la Figura 3.1 se puede observar que en esta instancia (rectas) no solo influye la familia en el comportamiento de la convergencia, sino que también influye desde donde comience el tiro (la convergencia es considerada como el instante de tiempo T para el cual para los instantes siguientes ya se tiene información suficiente para determinar, una vez calculado el t raíz tal que $x(t) = 125$ mediante método de la bisección instante a instante, valores de $y(t)$ razonables donde razonable significa que no varían mucho entre sí y está dentro del área del arco, por donde va a pasar la pelota con ese T (último instante, cantidad de puntos observados, en esta experimentación siempre se considera los instantes tales que $T \leq t$). Se hace esta salvedad en la experimentación por el caso que se observa al considerar el R2.tiro en esta familia de funciones. En particular, en este tiro, recién para $T = 18$ aproximadamente se obtienen resultados que podríamos considerar confiables sobre la posición final de la pelota. No ocurre esto en los demás tiros rectos, los cuales convergen desde mucho antes, es por esto que aclaramos que no solo importa la familia de función sino la pendiente y el punto inicial de la recta considerada. Esto nos da la pauta de que el método de interpolación mediante Splines Cúbicos no es muy preciso para la familia de funciones rectas.

En lo que refiere a las familias de funciones cúbicas y cuadráticas (Fig 3.3 y 3.2 respectivamente) podemos observar que el comportamiento de la convergencia es muy parecido, es decir, en cada familia parece haber un T para el cual, apartir del mismo, la exactitud con la cual se determina donde pasará la pelota por el eje y y cuando $x = 125$ comienza a crecer hasta que da por poco exacto en todos los casos cuanto más se va acercando la pelota al arco.

En el caso de las instancias con ruido el gráfico indica por sí solo que, al ser casi arbitrario el desplazamiento de la pelota, e impredecible, el polinomio calculado para la resolución del problema, nos brindará resultados absurdos instante a instante pero, como la precondition sobre las instancias de tiro tomadas es que la pelota siempre pase por el arco entonces siempre habrá una trayectoria en la cual se describa el movimiento final de la pelota hacia el arco, pero luego de pasar este puede continuar dándonos resultados absurdos pero esto ya no nos interesa. Es por esto que consideramos que el arquero solo se mueva en los instantes en los cuales no hay mucha variación respecto a la última predicción hecha dado el instante anterior y que siempre estén en el área del arco. En la figura 3.4, esos instantes son, aproximadamente, desde el instante $T = 15$, hasta el instante $T = 22$ (aproximadamente). Una aclaración que refiere a las escalas consideradas en cada gráfico de cada familia de funciones, es notable remarcar que las oscilaciones en las predicciones instante a instante de las rectas si bien difieren mucho y más en los primeros instantes de tiempo, es clave remarcar que siempre, en estas instancias en particular, lo hacen en un intervalo del eje y más reducido (en la Fig 3.1 las oscilaciones no escapan al intervalo $(-1500, 2500)$ en el eje y). En cambio en los gráficos de las instancias cuadráticas y las cúbicas este intervalo crece considerablemente, aproximadamente entre $(-6000, 8000)$ y entre $(-16000, 10000)$ respectivamente, lo cual nos da la pauta de que si bien, para los primeros instantes, la estimación de la posición final de la pelota en y es bastante desafortunada, la misma no presenta oscilaciones como es en el caso de las instancias rectas, pero, por otro lado, es claro que el pro de las rectas es que la variación en lo que refiere a distancia entre instante a instante, aún siendo oscilatorio, esta no escapa de un intervalo de resultado más chico que el de las otras instancias. Finalmente cabe destacar que en el caso de los tiros con ruido se puede esperar cualquier cosa en lo que refiere a este intervalo el cual encierra el valor donde, puede, o no, finalmente converger las

estimaciones de dicha posición final de la pelota, esto quiere decir que puede darnos cualquier resultado y de repente aparecer en el arco la pelota, complicando sumamente su predicción.

Sobre el Experimento 4 : Como se puede ver, los resultados cuando no desconsideramos los cálculos de la interpolación que se alejan mucho de la posición del arco, la efectividad en la atajada baja. Esto se debe a que como el error en el cálculo es muy grande, no vale la pena que el arquero se mueva de acuerdo a ese cálculo, pues se estaría moviendo a cualquier lado. Lo mismo sucede si no diferenciamos los casos en los que Splines calcula medianamente aceptable (la diferencia entre el cálculo anterior y el actual está entre 20 y 40) y cuando calcula muy bien (menos de 20 de diferencia). Creemos que esto es porque cuando la diferencia es de entre 20 y 40, lo que está sucediendo es que el cálculo está convergiendo a la solución final (aún no llegamos a la solución final) entonces lo mejor es movernos todo lo posible hacia esa posición, sea cual sea).

Además, uno esperaría que considerando sólo los puntos en los que $x > 125$, deberíamos tener más precisión en la atajada que considerando todos los puntos de la trayectoria (incluso los que pasan por atrás del arco), pero no pudimos observar que esto sucediera. De hecho, la precisión en la atajada no se modificó y el arquero atajó los mismos tiros. Se podría decir que, teniendo en cuenta los otros criterios del arquero (con sus parámetros correspondientes, elegidos por nosotros arbitrariamente) logramos indirectamente que el arquero se mueva de forma muy similar considerando más puntos. Entonces, en cuanto a esto no podemos sacar conclusiones pues en principio no queríamos que el arquero considerase los puntos de la trayectoria de la pelota cuando esta ya pasó por el arco para el cálculo de la función interpoladora.

5. Conclusiones

Sobre el Experimento 1 : Como se observó en la discusión, el método de interpolación que implementamos efectivamente hace lo que nosotros pretendemos que haga, que es que progresivamente, al considerar cada vez más puntos de una determinada trayectoria, vayamos obteniendo cada vez un valor de posición final de la pelota más cercano al de la realidad, y que para sólo cierta fracción de la totalidad de los puntos de la trayectoria que tenemos a nuestra disposición, seamos capaces de conocer con una precisión aceptable (± 7 unidades como máximo), cuál será la posición final de la pelota cuando hayamos considerado la totalidad de los puntos que sabemos recorrió la pelota.

Sobre el Experimento 2 : En este caso, como se predijo en la hipótesis, observamos que las curvas de recorrido de la pelota que obtuvimos con nuestro método de interpolación se correspondieron, para ciertos valores de T (cantidad de puntos del tiro consideradas), con las curvas características que se suponen que deberían tener los tiros (respectivamente rectas, cuadráticas y cúbicas). Luego, se puede decir que el método de interpolación de puntos que implementamos nos devuelve los resultados esperables, si consideramos que efectivamente los tiros que testeamos (provistos por la cátedra) efectivamente tienen distribución recta, cuadrática y cúbica respectivamente.

Sobre el Experimento 3 :

Como conclusión de este experimento, podemos decir que el método de interpolación mediante el Spline Cúbico natural, converge a un resultado aceptable y razonable en lo que refiere a las predicciones de la posición final de la pelota cuando la misma se encuentra llegando al arco. Concluimos, además, que el comportamiento de convergencia varía según la familia de funciones a la cual se le aplica dicho método. Más aún, al menos en el caso de las rectas, por lo que pudimos ver, también es importante el punto de partida de la pelota. En el caso de las funciones cuadráticas y cúbicas, el método funciona bastante bien, mejor aún en las cúbicas, creemos que esto es así justamente porque estamos intentando aproximar las mismas con polinomios de grado 3 los cuales se ajustan mejor a la trayectoria descrita por las instancias de esa familia de funciones. En el caso de las instancias donde la trayectoria presenta ruido, podemos concluir que el resultado obtenido con el Spline no nos es muy útil al menos hasta que la pelota está lo suficientemente cerca del arco, sin considerar los instantes de tiempo anteriores que nos devuelven resultados absurdos; donde, es a partir de dicho instante cercano al arco que la pelota toma su trayectoria final en dirección al mismo y al menos el arquero obtiene una información en la cual confiar. Este análisis deja en claro que es necesario generar distintos tipos de criterios para el arquero, dependiendo de la familia de funciones a considerar, hasta incluso de la instancia en particular que se considera de cada una de estas familias, es decir que el Spline por sí solo no alcanza para dar a priori una predicción certera para determinar como se va a desplazar al arquero para atajar la pelota.

Sobre el Experimento 4 : Podemos concluir que los criterios que utilizamos para que el arquero ataje son, junto con los parámetros que consideramos arbitrariamente, aceptablemente buenos a la hora de hacer que el arquero ataje, mejorando el porcentaje de atajadas del arquero lo máximo que pudimos. Se podría decir que estos criterios, combinados con el método de interpolación mediante Splines Cúbicos son bastante buenos. Más aún, teniendo en cuenta que en los tests en los que utilizamos tiros con ruido, el arquero tuvo porcentajes de eficiencia en la atajada muy similares a los obtenidos para los otros tiros, normales.

Conclusión final del informe: Como conclusión final del informe podemos objetar que el método de interpolación segmentaria mediante Spline Cúbico natural es una buena forma, dada una cantidad suficiente de puntos observados y un comportamiento razonable de la función, para aproximar una función mediante interpolación, en particular polinómica. De todos modos hay que analizar de forma particular el problema que se intenta resolver con este método, ya que, si bien es útil y necesario, puede no ser suficiente para, por ejemplo en este problema planteado, resolver los movimientos del

arquero para atajar la pelota cuando la misma ingresa en el arco, el cual necesitará, además de la predicción hecha por el polinomio obtenido mediante el método, otros criterios de decisión fundamentales.

Método alternativo: Presentaremos a continuación un método secundario nombrado en la introducción, el cual no implementamos en nuestro código pero que también sirve para resolver el problema en cuestión, este es Cuadrados Mínimos. Mostraremos el problema a tratar y su solución mediante este método, como no se pierde generalidad presentaremos al mismo aplicado al problema bidimensional.

Sea $(x_k, y_k) k = 1 \dots n$ un conjunto de n datos que conforman una muestra observada y sea $f_j(x_k) j = 1 \dots m$ una base de m funciones linealmente independiente en un espacio de funciones. Nosotros queremos hallar una función $f(x)$ tal que sea una combinación lineal de las funciones pertenecientes a la base tomada. De modo que $f(x_k) \approx y_k$. Es decir:

$$f(x) = \sum_{j=1}^m c_j f_j(x)$$

Se intentará encontrar los m coeficientes que hagan que la función final que aproxima $f(x)$ de la mejor aproximación para todos los puntos (x_k, y_k) . Dicha "mejor aproximación" da lugar a muchos criterios a considerar, pero, en general se utiliza el que minimiza la acumulación de errores individuales (de cada punto) sobre el conjunto total. El error individual para cada punto se define como (cabe destacar que nos interesa minimizar el error en todo el conjunto de aproximación) :

$$e_k = y_k - f(x_k)$$

Hay varias maneras de representar dicho error, un par de ellas son:

$$\begin{aligned} \text{Error Medio } E_m(f) &= \frac{\sum_{k=1}^m |e_k|}{n} \\ \text{Error Cuadrático Medio } E_{cm}(f) &= \sqrt{\frac{\sum_{k=1}^m (e_k)^2}{n}} \end{aligned}$$

Para resolver este problema la solución obvia sería derivar el Error Medio e igualarlo a 0 para encontrar el punto crítico y determinar el mínimo. El problema surge en que el módulo es decir $|e_k|$ de la expresión del Error Medio, no es derivable; y, es por esto que se decide considerar el Error Cuadrático Medio que si es derivable, en particular, es equivalente tomar y minimizar el radicando de la expresión E_{cm} .

Finalmente, si generalizamos en la dimensión del problema, lo que estamos resolviendo es lo siguiente:

$$\min_{a_1, \dots, a_n} \sum_{i=1}^n (a_1 \varphi_1(x_i) + \dots + a_n \varphi_n(x_i) - y_i)^2$$

Donde $a_1 \varphi_1(x_i) + \dots + a_n \varphi_n(x_i)$ será la combinación lineal de las funciones tomadas de la base de funciones que minimiza el Error Cuadrático Medio.

Si lo vemos de forma matricial podríamos escribir la generalización de la siguiente manera:

$$A = \begin{pmatrix} \varphi_1(x_1) & \dots & \varphi_n(x_1) \\ \varphi_1(x_2) & \dots & \varphi_n(x_2) \\ \vdots & & \vdots \\ \varphi_1(x_n) & \dots & \varphi_n(x_n) \end{pmatrix} a = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} b = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{pmatrix}$$

Esto se traduce, en notación de normas matriciales, en:

$$\min_a \|Aa - b\|_2^2 = \min_a (Aa - b)^t (Aa - b) = \min_a (aA^t - b^t)(Aa - b) = \min_a (aA^t Aa - x^t A^t b - b^t Aa + b^t b)$$

Si derivamos en a (incógnita) el último resultado e igualamos a 0 estaríamos encontrando el mínimo que necesitamos, entonces:

$$\min_a \|Aa - b\|_2^2 \equiv \text{encontrar } a \in \mathbb{R}^n \text{ tal que } 2A^t Aa - 2A^t b = 0 \equiv \text{encontrar } a \in \mathbb{R}^n \text{ tal que } A^t Aa = A^t b$$

Este último resultado que se obtiene de la forma de escribir la norma y derivarla se conoce como Ecuaciones Normales (existen otros métodos para encarar el problema de encontrar el $a \in \mathbb{R}^n$ que minimice la expresión, se puede utilizar la descomposición SVD o QR de A). AA^t es simétrica definida positiva si las columnas de A son Linealmente Independientes, sino es simétrica semi definida positiva. En este caso nuestra matriz A tendrá siempre rango máximo, por lo que AA^t es siempre simétrica definida positiva en este problema.

Para resolver este sistema final planteado, bastaría triangular la matriz mediante triangulación gaussiana. La solución va a verificar las Ecuaciones Normales, pero podría no verificar $Aa = b$, ya que, uno, al encontrar la solución de cuadrados mínimos, está encontrando el $a \in \mathbb{R}^n$ tal que al hacer Aa nos da la componente b en la imagen de A, que minimiza dicha expresión.

A continuación se muestra la prueba de que dicho $a \in \mathbb{R}^n$ que se encuentra, que nos da la proyección ortogonal de b en la imagen de A, es el que minimiza la expresión planteada del problema que intentamos resolver:

Prueba: Escribimos $b = b_1 + b_2$ donde $b_1 \in Im(A)$ y $b_2 \in Im(A)^\perp$

Como dijimos antes, buscamos minimizar $\|Aa - b\|_2^2 = \|Aa - (b_1 + b_2)\|_2^2 = \|(Aa - b_1) - b_2\|_2^2$, como $(Aa - b_1) \in Im(A)$ y $b_2 \in Im(A)^\perp$ entonces, por pitágoras, la expresión es igual a $\|(Aa - b_1)\|_2^2 + \|b_2\|_2^2$. Ambas expresiones, por ser normas, son mayores o iguales que 0, entonces, para minimizar esta expresión al máximo, necesitamos encontrar dicho $a \in \mathbb{R}^n$ que nos da la proyección ortogonal de b en la Im(A) (es decir, b_1), de manera tal que $\|(Aa - b_1)\|_2 = 0$. El mismo es el que verifica, como se dijo, $AA^t a = A^t b$, pero podría no verificar $Aa = b$.

6. Apéndices

6.1. Enunciado

Laboratorio de Métodos Numéricos - Primer Cuatrimestre 2014 Trabajo Práctico Número 3: Hay que poner un poquito mas de esfuerzo...

Introducción

Debido al éxito conseguido con el sistema de reconocimiento de barras, el Comité Ejecutivo de FIFA® ha aceptado otorgar a la Selección Argentina el beneficio de reforzar el equipo mediante el reemplazo de alguno de los *muertos* que se encargan de los aspectos defensivos del equipo por un robot, siempre y cuando siga las Tres Leyes de la Robótica [?].¹ Ante esta posibilidad, Sabella no dudó y se ha contactado con el Equipo de Desarrollos de Métodos Numéricos para, esta vez, colaborar para traer *la tan ansiada copa* a la Argentina luego de 28 años, desde aquella alegría del Diego. El análisis preliminar determinó que se deben estudiar y adaptar las técnicas utilizadas habitualmente en el *fútbol de robots* y desarrollar una herramienta que se llamará *Localized Trajectory Approximator* (LTA).

Uno de los problemas más importantes a resolver en este contexto es predecir con la mayor anticipación posible la posición futura de la pelota en función de su posición en el pasado reciente. Sobre la base de estas predicciones se coordinan los movimientos de los jugadores de campo y, en el caso que nos ocupa ahora, la posición del arquero cuando existe peligro de gol. Cuando la pelota se dirige hacia nuestro arco, es muy importante que ubiquemos el arquero en la posición exacta en la que la pelota cruzará la línea de gol, de manera que pueda interceptarla y evitar la caída de nuestra valla. El sistema de control de cada equipo suministra información en pasos discretos. En cada paso nuestras cámaras de video determinan la posición de la pelota, y debemos indicarle la acción a seguir al arquero.

Los postes del arco están ubicados en coordenadas fijas, y la línea de gol es el segmento entre estos dos puntos. Se marca un gol cuando la pelota cruza este segmento. Vistos desde arriba, la pelota es un círculo de radio determinado y el arquero se representa mediante un segmento paralelo a la línea de gol, ubicado sobre la misma. Inicialmente el arquero se encuentra en punto del plano, sobre la línea de gol, y en cada paso se le indica al arquero qué acción debe tomar. Las posibles acciones en cada instante de tiempo son dos: moverse hacia alguno de los lados, izquierda o derecha, una cantidad de pasos (acotada por un valor máximo μ), o quedarse quieto y no hacer nada.

Formalmente, consideraremos un horizonte discreto de tiempo $0, 1, \dots, T$ en el cual se realiza el disparo y describiremos la trayectoria de la pelota mediante una función $p : \mathbb{R} \rightarrow \mathbb{R}^2$, $p(t) = (x(t), y(t))$ que permite describir la posición de la pelota en función del tiempo. Tanto el arco, como los jugadores rivales y nuestro arquero se ubican en posiciones que representaremos con coordenadas (x_i, y_i) en el plano \mathbb{R}^2 . En particular, nuestro arquero sólo se moverá sobre el eje y , manteniendo fija su coordenada x en la línea de gol. El objetivo del trabajo es en cada instante de tiempo reportar un valor z , $-\mu \leq z \leq \mu$ que indique cuánto y hacia dónde se desplaza el arquero respecto de la posición actual.

Debido a la tecnología utilizada para la detección de la pelota, un problema fundamental que debe enfrentarse es la presencia de ruido en las mediciones de la posición de la misma. El sistema de visión está sujeto a vibraciones, golpes y errores de captura de datos, que hacen que las mediciones de la pelota sufran errores. Por otra parte, la pelota no siempre viaja hacia el arco en línea recta sino que puede describir curvas más o menos complicadas, dependiendo del “efecto” dado por el jugador al momento de impactar la pelota y de posibles curvaturas en la superficie del campo de juego.

Además de la información del arquero y la pelota, contaremos también con las posiciones de los jugadores rivales en la cancha. El objetivo de considerar esta información adicional radica en simular una situación de juego real. Para simplificar levemente el alcance en esta etapa asumiremos que la posición de los jugadores no varía con el tiempo, aunque sí pueden intervenir en el juego y desviar la

¹Se está considerando *darle hacha* a la rodilla de Neymar Jr. o de Alexis Sánchez en un eventual cruce con las selecciones hermanas.

pelota con un remate al arco o un pase a otro jugador.

Enunciado y competencia

Se pide implementar un programa en **C** o **C++** que resuelva el problema de determinar la posición del arquero utilizando técnicas y métodos vistos en la materia. Para ello, dada la posición inicial del arquero, los límites del arco, las posiciones de los jugadores del equipo rival y la evolución de la posición de la pelota (o su estimación) en distintos instantes de tiempo, deberá determinar qué acción debe realizar el arquero en cada momento con el objetivo final de atajar el disparo. Si bien los datos pueden tener ruido debido al sistema de captura, asumiremos que la última medición carece de error y representa la posición efectiva de la pelota.

El trabajo constará de al menos un método que pueda ser utilizado para predecir la posición futura de la pelota, implementado íntegramente por el grupo (incluyendo sus métodos de resolución). En caso de considerar más de una alternativa, es posible desarrollar los restantes métodos utilizando librerías públicas (por ejemplo, *boost* [?]). En todos los casos es imprescindible incluir el desarrollo una descripción detallada de los mismos, justificando adecuadamente su elección. Además de su explicación, es obligatorio que todos los métodos propuestos sean debidamente evaluados y analizados experimentalmente, mediante la formulación de hipótesis, realización de experimentos y el posterior análisis de los resultados obtenidos en los mismos, buscando caracterizar y justificar su comportamiento en función del problema a resolver.

En el contexto del trabajo, los arqueros propuestos por los distintos grupos participarán, junto con un arquero propuesto por la cátedra, en una competencia que se llevará a cabo el día 27 de Junio de 2014, a las 18:00 hs, luego de la devolución de los trabajos corregidos. El formato de la competencia, las instancias de prueba y los premios serán informados durante los primeros días de Junio. Es importante destacar que la aprobación del Trabajo Práctico es independiente de la competencia.

Para simplificar la implementación, el archivo de entrada contará con toda la información del disparo, aunque para tomar la decisión en el instante actual no se podrá utilizar información futura. Aquellos grupos que violen esta regla serán automáticamente reprobados y descalificados de la competencia.

Programa y formato de archivos

El programa a desarrollar deberá tomar por línea de comandos un archivo de entrada con la descripción del tiro y generar un archivo de salida, cuyo nombre será también indicado mediante un parámetro, con los movimientos a realizar por el arquero en cada instante de tiempo.

El archivo de entrada seguirá el siguiente formato:

- La primera línea contendrá la posición inicial del arquero en y , y luego las coordenadas que definen los límites del arco, también sobre el eje y . Se asume que la posición en x del arquero y de la línea de gol son las mismas: $x = 125$. Finalmente estará μ , la cota sobre el máximo desplazamiento que puede realizar el arquero en un instante de tiempo.
- Luego se muestra la secuencia de posiciones en \mathbb{R}^2 , una por línea, que toma la pelota para los instantes de tiempo $0, 1, \dots, T$.

A continuación se muestra un archivo donde el arquero comienza en la posición $(125, 375)$, donde el arco tiene su límite superior en la coordenada $(125, 385)$ y la inferior en $(125, 315)$. Recordar que el arquero siempre se encuentra sobre la línea de gol. Finalmente, $\mu = 4$, y luego se tiene una secuencia de 24 posiciones con la definición del tiro para los instantes de tiempo $0, 1, \dots, 23$.

```
375 315 385 4
468.0 522.0
450.0 512.0
432.0 502.0
414.0 492.0
396.0 482.0
378.0 472.0
360.0 463.0
342.0 453.0
324.0 443.0
306.0 433.0
288.0 423.0
270.0 413.0
252.0 403.0
234.0 393.0
216.0 383.0
198.0 374.0
180.0 364.0
162.0 354.0
144.0 344.0
126.0 334.0
108.0 324.0
90.0 314.0
72.0 304.0
54.0 295.0
```

El archivo de salida deberá contener una instrucción por línea, correspondiente a la acción que realiza el arquero en el instante $0 \leq t \leq T$. Un posible resultado para el tiro descrito anteriormente es el siguiente, donde el arquero no realiza ningún movimiento salvo en el último instante donde se mueve a la posición $(125, 371)$.

```
0
0
```


7. Referencias

- R. Burden y J.D.Faires, Análisis numérico, International Thomson Editors, 1998
- D. Watkins, Fundamentals of matrix computations, John Wiley & Sons, 1991