

Implementación de un algoritmo para el seguimiento del sol.

¹Roldán Rubio, Homero

(¹) Gerencia de Posgrado y Formación Continua, CIATEQ A.C.
Av. del Retablo 150 Col. Constituyentes Fovissste. Querétaro, Qro. México. 76150
Teléfono: (442) 211.2600 Ext.2555 y 2673

Resumen

En este artículo se propone un nuevo punto de vista para realizar el seguimiento del sol, en el problema de parametrización de curvas complejas dados un número finito de puntos conocidos se busca que una curva suave y continua pase por dichos puntos, En el análisis numérico existen diversas herramientas una de ellas es la B-spline, que en base a un vector de nodos, nos permite optimizar la spline en base a los nodos, Se propone un algoritmo genético para optimizar la trayectoria por medio de la B-Spline.

Palabras clave: Trayectoria, Sol, Optimización.

1. Introducción

El seguimiento del sol tiene aplicaciones en la vida cotidiana desde el principio de la humanidad donde era ocupado para medir el tiempo, hasta la actualidad que es muy importante para el uso de las ahora conocidas como energías renovables o energías verdes.

La energía del sol es limpia, renovable y disponible en cada parte del mundo. Las necesidades de energía y los costos se han incrementado en los últimos años y la naturaleza se daña durante el proceso de la producción, con el protocolo de Kioto se ha propuesto la disminución de contenidos y la migración a las energías verdes [3].

Los seguidores del sol en la actualidad son utilizados en la recopilación de energía solar, para su eventual uso en diversos ramos, principalmente eléctrica como Topaz Solar Farm en el condado de San Luis Obispo California o uno de los proyectos más ambiciosos OptiSolar Farms Canadá Inc en Ontario Canadá de 365 hectáreas, en los proyectos de esta índole se hace énfasis en la optimización de la recopilación de energía.

La diferencia en la eficiencia de la captación de energía es proporcional a si se realiza el seguimiento del sol (Figura 1) y cuan preciso es el mismo [3]. Los algoritmos que utilizan en la actualidad brindan certeza en el posicionamiento del sol y busca la mayor captación de energía con la menor cantidad de cálculos posibles.

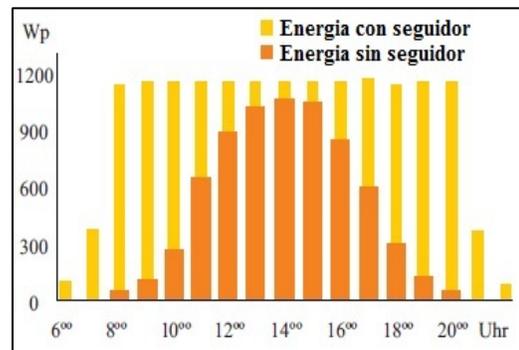


Fig. 1. Eficiencia con seguidor Sousse, Tunes.

2. Fundamentos

Se describirán las herramientas necesarias para la comprensión del artículo

2.1 Mecánica del Sistema.

Se desea que el sistema cumpla con ciertas condiciones es decir que pueda realizar ciertos movimientos y alcanzar ciertas velocidades para poder realizar el seguimiento del sol de manera óptima.

Se desea que θ_1 que es ángulo azimutal esté en un rango de 50° a 310° pues en la ciudad de Querétaro de Arteaga nunca amanece antes de las 5:00 y nunca oscurece después de las 21:00 horas. En el caso de la elevación θ_2 se pide que se tenga al menos un

rango de -30° a 110° dando una amplitud suficiente y necesaria para la posición geográfica del municipio..

Las ecuaciones de la cinemática están dadas por (1) y (2) [4].

$$\theta_1 = \tan^{-1} \frac{p_y}{p_x} \quad (1)$$

$$\theta_2 = \tan^{-1} \frac{\sqrt{p_x^2 + p_y^2}}{d_1 - p_z}, \quad (2)$$

lo cual expresa variación máxima en la velocidad de más de 10.6° grado/segundo en el eje azimutal y 0.23° grado/segundo en el ángulo de elevación.

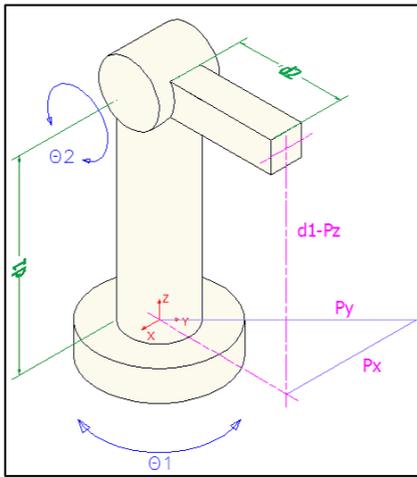


Fig. 2. Sistema con dos grados de libertad.

2.2 SPA (Solar position Algorithm)

EL algoritmo SPA es recomendado para el cálculo de la posición del sol [5] con respecto al tiempo y la localización en la tierra, haciendo uso de los algoritmos VSOP87 Variations Seculaires des Orbites Planetaires [6] donde se describen las órbitas de los planetas y como estas se afectan entre sí, dando una forma de localización de los astros con mayor exactitud dentro del sistema solar.

Este SPA nos da una precisión de $\pm 0.0003^\circ$ como desventaja de este algoritmo se tiene el tiempo máquina que consume en la realización del cálculo de cada uno de los valores, para un tiempo $t \in (0,24)$ dado en cualquier día del año. Con la siguiente estructura:

```
location.longitude = -100.40844;
```

```
location.latitude = 20.59807;
location.altitude = 1830.14;
time.year = 2013;
time.month = 05;
time.day = 06;
time.hour = 5;
time.min = 00;
time.sec = 00;
time.UTC = -6;
```

El principal problema de este algoritmo es que ocupa demasiado tiempo máquina para calcular una posición es decir θ_1, θ_2 , el algoritmo implementado en el software de Matlab 7.9.0 R2009b en una computadora Toshiba con procesador Intel i5 CPU M 450 @ 2.40GHz, con 4.00 GB en RAM el cálculo se lleva a cabo en aproximadamente 0.85 a 0.25 segundos y en los seguidores se requiere un reposicionamiento frecuente, para optimizar la captación de la energía. Si se quisiera hacer un reposicionamiento cada 7.5 minutos desde las 5 horas a las 20 horas se requerirían 121 cálculos.

En vista de esto se utilizará una interpolación para aproximar los valores y no tener que estar llamando la función si requiere un posicionamiento más frecuente dígame cada 10 segundos o menos.

Grafica de la posición del sol para el día 6 de mayo del 2013 tabla 1 y figura

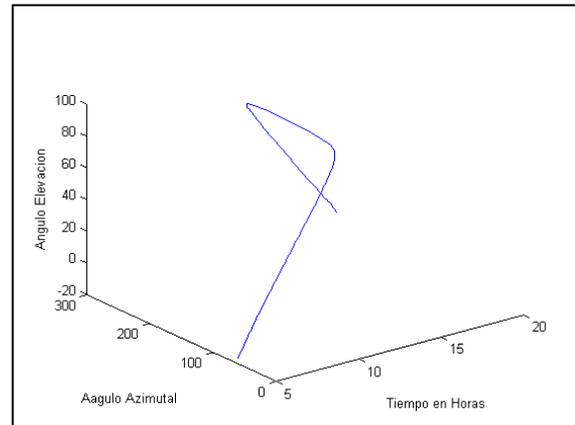


Fig. 3. Sistema con dos grados de libertad.

Tiempo Hora del día	Elevación en grados	Azimutal en grados
5:00	-15.834147	64.9004918
7:30	17.622105	78.3148028
10:00	52.427366	89.1329442
12:30	85.677476	153.025856
15:00	56.268935	269.542670
17:30	21.404011	280.624469
21:00	-24.740617	300.88940

Tabla. 1. Sistema con dos grados de libertad.

2.3 B-spline

La interpolación es una rama del análisis numérico en la cual se representan curvas dado un conjunto de punto conocido por los que pasa la curva original (Puntos de control) [1][2] que se desconoce, otra de sus aplicaciones es cuando la curva es muy difícil de calcular y solo se puede obtener un numero finito de muestras, y se necesita aproximar el comportamiento de la curva entre los puntos obtenidos.

La B-spline tiene la ventaja con respecto a otro tipo de splines de que puede representar curvas de manera suave y hasta n-2 veces diferenciable, dado el orden de la base n.

Se puede construir una curva polinomial a trozos de la siguiente forma

$$Q(u) = \sum_{j=0}^n P_j B_{j,d}(u) \quad (3)$$

Con $t_{d-1} \leq u \leq t_{n+1}$ donde P_j es un punto de control j es el índice de los puntos de control $n + 1$ es el número de puntos de control, d es el número de puntos de control dentro de un segmento de control, esta cantidad nos da el grado de nuestro polinomio. El valor de $n \geq d$, t_j es un valor del vector Nudo. Un conjunto de ellos nos da un vector de nodos t .

Las funciones $B_{j,d}(u)$ forman una base en la cual se construye nuestra Spline.

Si $d=1$

$$B_{j,1}(u) = \begin{cases} 1 & \text{si } t_j \leq u < t_{j+1} \\ 0 & \text{en otro caso} \end{cases} \quad (4)$$

Y si $d>1$

$$B_{j,d}(u) =$$

$$\left(\frac{u-t_j}{t_{j+d+1}-t_j} \right) B_{j,d-1}(u) + \left(\frac{t_{j+d}-u}{t_{j+d}-t_{j+1}} \right) B_{j+1,d-1}(u) \quad (5)$$

En las dos ecuaciones anteriores se puede observar que el vector Nodos afecta la B-spline. Es por ello que se tiene diferencia cuando hay un vector Nodos uniforme y uno que no lo es [7].

Cabe notar que la función $B_{j,d}(u)$ es llamada función de Bezier de orden d y es una función de clase C^{d-2} continua, lo cual no permite obtener con la misma curva aproximaciones de la velocidad y de la aceleración.

3. Metodología

En el caso de este trabajo se propone el uso de la interpolación B-spline [1] para interpolar de manera óptima la posición del sol, con respecto a un punto fijo de la tierra sin la necesidad de mandar llamar la función para el cálculo de la posición del exacta sol.

Para ellos se propone como en [1] el uso de un algoritmo genético para optimizar el vector de k-not que tiene injerencia en el cálculo del de interpolación B-spline,

En el algoritmo genético se toman 6 vectores Nodos para que con cada uno de ellos se calcule su respectivo B-spline Se realiza una función de comparación entre las B-spline que generan vectores de información los cuales son de longitud superior a 1000 por lo tanto se toma un muestreo dada la longitud del vector.

Nuestra función de peso poder estar dada de dos maneras de forma principal la primera es optimizar relación del el eje elevación y la segunda con respecto el eje azimutal. Por lo tanto nuestro algoritmo queda de la siguiente forma:

Paso 1 Tomar 6 vectores uno de ellos uniformemente distribuido y los demás de forma aleatoria (V1, V2, V3, V4, V5, V6) y un número n' que es la cantidad generaciones.

Paso 2 Tomar mediante la función de peso los dos mejores vectores (V1', V2') dadas sus B-spline.

Paso 2.1 Si $V1' < \epsilon$ terminar el proceso entonces V1' es el vector que cumple con las condiciones deseadas.

Paso 3 Generar 6 nuevos vectores

Paso 3.1 $V1=V1'$ & $V2=V2'$

Paso 3.2 La parte entera de $V3=(V1+V2)/2$

Paso 3.3 El $V4$ se generará como la mezcla de $V1$ & $V2$ con la siguiente regla: $V4(i)=V1(i)$ si i es múltiplo de 2 y $V5(i)=V2(i)$ si i no es múltiplo de 2

Paso 3.4 El $V5$ se generará como la mezcla de $V1$ & $V2$ con la siguiente regla: $V5(i)=V1(i)$ si i no es múltiplo de 2 y $V5(i)=V2(i)$ si i es múltiplo de 2

Paso 3.5 $V6$ Se generará de manera aleatoria.

Paso 4 $n=n+1$

Paso 5 Volver al Paso 2 a menos que $n=n'$, entonces el mejor vector encontrado es $V1'$.

De este algoritmo se calculó la información y se obtuvieron los siguientes resultados

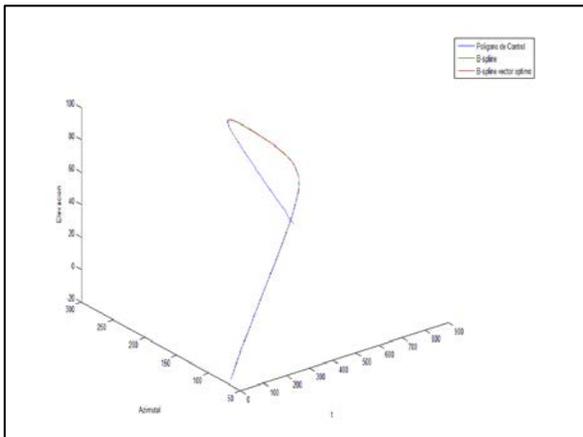


Fig. 4. Movimiento del sol para el 06/05/2013, aproximación por medio B-spline uniforme y B-spline optimizada por AG con 200 generaciones

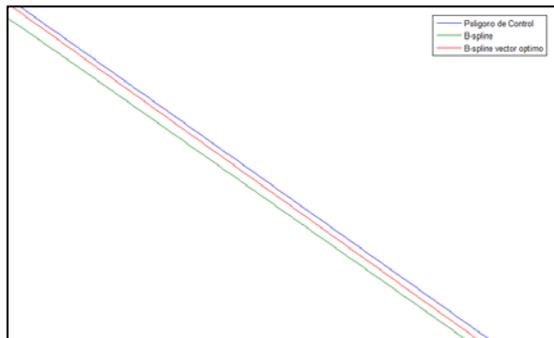


Fig. 5 Movimiento del sol para el 06/05/2013, aproximación por medio B-spline uniforme y B-spline optimizada por AG con 200 generaciones con un zoom de 450%

Como se puede ver en la Figura 4 y 5 la curva optimizada por un algoritmo genético está por encima de la b-spline con el vector de nodos uniforme, el error acumulado en el eje de elevación es de 0.239813 grados a lo largo de 16 horas y en el eje azimutal es de 0.69581 grados en el mismo tiempo, usando como función de ponderación el error en el eje de elevación. En la figura 6 y 7 la curva optimizada por un algoritmo genético está por encima de la B-spline con el vector de nodos uniforme, el error acumulado en el eje de elevación es de 0.339813 grados a lo largo de 16 horas y en el eje azimutal es de 0.5521773 grados en el mismo tiempo, usando como función de ponderación la error en el eje azimutal.

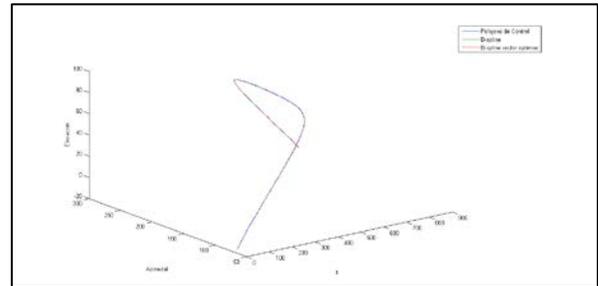


Fig. 6. Movimiento del sol para el 06/05/2013, aproximación por medio B-spline uniforme y B-spline optimizada por AG con 200 generaciones

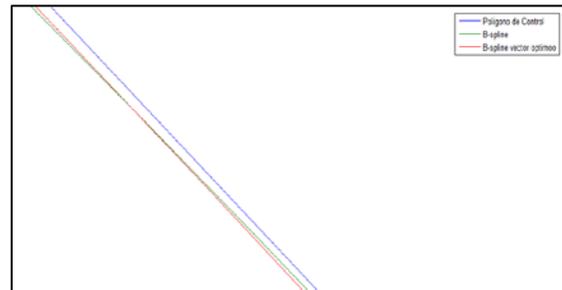


Fig. 7. Movimiento del sol para el 06/05/2013, aproximación por medio B-spline uniforme y B-spline optimizada por AG con 200 generaciones con un zoom de 450%

Cuando se realiza el cálculo cuando el número de generaciones es igual a 300 se tiene que la función de ponderación está dada por el error en el eje de elevación, teniéndose que el error en el eje de elevación es de 0.213537 y en el eje azimutal de 0.587172.

Si se tomara la función de peso con respecto al eje azimutal entonces el error en el eje azimutal sería de 0.45072 y 0.26758 para el de elevación.

4. Conclusiones

Se puede ver en base a los resultados que se puede aproximar a la trayectoria del sol con un rango aceptable de error a lo largo de un periodo 16 horas, pues con un error de 5° la eficiencia en la captación de energía solo se reduce en un 0.4% en heliostatos, se propone calcular el vector de nudos con días de anticipación y solo incluir dentro del controlador la función de la B-spline para reducir el tiempo de procesamiento.

En trabajos futuros se buscará compara con otros tipos de spline Cúbica y lineal o formas para aproximar funciones como la wavelets, u ocuparlo para el diseño o cálculo de trayectorias óptimas que pasen por un número finito de punto.

Se busca generalizar el algoritmo para la mayor cantidad de situaciones posibles es decir abarcar la mayor cantidad de situaciones en las cuales se pueda presentar un erro ya sea por refracción o movimiento del dispositivo.

Referencias

- [1] GOLDENTHAL, R. *Spline Curve Approximation and design by optimal control, over the knots using genetic algorithms*, International Congress on Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems, EUROGEN 2003 pp 1-18.
- [2] GOLDENTHAL, R. *Spline Curve Approximation and Design by Optimal Control Over the Knots*, Computing 72 pp 53-54, Springer-Verlag 2004.
- [3] RHIF, A. *A Position Control Review for a Photovoltaic System: Dual Axis Sun Tracker*, IETE Tech Rev 2011 28 pp 479-85.
- [4] LOZANO, C. *Diseño Conceptual de un Robot de Dos Grados de Libertad*, Instituto Tecnológico de Celaya, Tesis de grado de maestría, Diciembre 2012.
- [5] REDA, I. *Solar Position Algorithm for Solar Radiation Applications*, Solar Energy 76 (2004) 577-589.
- [6] BRETAGON, P. *Planetary Theories in rectangular and spherical variables VSOP87 Solution*, Astron, Astrophys 202, pp 309-315 1988.
- [7] TAN, R., *B-Spline curves, A Lecture Note of 3D Modeling Course*, Universititet Utrecht, 2011-2012.