Implementación de Algoritmos para la Búsqueda de Sucesiones Sonares y Sucesiones PPM

Eliseo Gallo Albarracín
DBA Gerencia en Sistemas de Información
Universidad del Turabo, Puerto Rico
Docente-Investigador Grupo SIGMMA,
Universidad Santo Tomás Bucaramanga, Colombia
eligallo@hotmail.com

Carlos A. Gómez Olejua
M.Sc. Física
Universidad Industrial de Santander
Docente-Investigador Grupo SIGMMA,
Universidad Santo Tomás, Bucaramanga, Colombia
carlosarnulfo@gmail.com

Resumen — En un salto de frecuencia de un sistema de radar, la señal se compone de una o más frecuencias elegidas de una posible combinación de m frecuencias disponibles para la transmisión en n intervalos consecutivos de tiempo. Esta señal puede ser representada por una matriz de $m \times n$ de 0's y 1's, donde es necesario que en cada columna contenga exactamente un 1. Cuando la señal es reflejada hacia el observador, esta se desplaza en el tiempo y frecuencia. La cantidad de estos movimientos (desplazamientos) se pueden utilizar para determinar la distancia y velocidad. La cantidad de estos saltos, a su vez, se determinan mediante la comparación de todos los turnos de una réplica de la señal transmitida con la señal recibida. Esto es equivalente a contar el número de coincidencias de 1's en una versión desplazada de la matriz de 0's y 1's que representa la señal. El número de coincidencias, como una función de cambios en el tiempo y la frecuencia se llama la función de "auto-correlación". Una matriz sonar es un modelo $m \times n$ que tiene a lo más una coincidencia con su función de auto-correlación. En un entorno de múltiples objetos, un patrón se envía para cada objetivo. En este trabajo se presentan algunos métodos que generan secuencias de sonares para el reconocimiento de objetivos múltiples, y también se hace mención a algoritmos de búsqueda para la mismas; como caso particular, se expone el uso de la técnica "*backtracking*" para hacer una búsqueda exhaustiva para encontrar secuencias de sonares.

Palabras clave— Algoritmo, *backtracking,* complejidad, Costas, Sucesiones, Sonares, PPM.

Abstract— In a frequency hopping radar system, the signal consists of one or more frequencies chosen from a set of m available frequencies for transmission at each of a set of n consecutive time intervals. Such a signal can be represented by an m X n matrix of 0's and 1's in which each column contains exactly one 1. When the signal is reflected back to the observer, it is shifted in both time and frequency. The amounts of these shifts can be used to determine both distance and velocity. The amounts of shifts, in turn are determined by comparing all shifts of a replica of the transmitted signal with the signal received. This is equivalent to counting the number of coincidences of 1s in a shifted version of the matrix of 0's and 1's that represents the signal. The number of such hits as a function of shifts in time and frequency is called the auto-correlation function. A sonar array is an $m \times n$ pattern which has at most one hit in its auto-correlation function. In a multiple target environment, one pattern is sent for each target. In this work we study algorithms that generate by exhaustion sonar type sequences for one and multiple target recognition, and particularly we expose use of *backtracking* technique to find sonar sequences.

Keywords **—** *Algorithm, backtracking, complexity, Costas, sonar sequences, PPM sequences.*

I. INTRODUCCIÓN

Las Secuencias de Costas y Secuencias Sonares fueron introducidas en [1] y [3] (respectivamente) para responder al siguiente problema: "Un objeto se está moviendo a cierta distancia de un observador, y él quiere saber efectivamente la distancia del objeto y su velocidad". La solución de este problema es obtenida usando el efecto Doppler, el cual dice lo siguiente: cuando un brinco de señal se mueve respecto a un blanco, su frecuencia cambia en directa proporción a la velocidad. En otras palabras, si el observador envía una señal hacia un blanco en movimiento, el cambio entre la frecuencia original y la frecuencia que retorna, permite calcular la velocidad del blanco y cuánto tiempo gasta en ir y volver la señal para así determinar la distancia.

Sin embargo, ya que el mundo es ruidoso, una señal que es enviada puede no retornar. Consecutivamente, usted envía l señales, cada una con una frecuencia de un intervalo de 1 hasta n en rangos de tiempo que varían desde 1 hasta l. En el caso donde $l = n$, donde en cada unidad de tiempo fue enviada una y sólo una secuencia, y esta clase de códigos es genéricamente llamada código "Frecuencia Hop" (FH), en el caso general le llamamos, Secuencia Sonar. Una vez el modelo de señales ha sido recuperado, la velocidad y la distancia pueden ser determinadas como se mencionaba arriba. Por otra parte, si alguna señal no regresa, algunas ambigüedades aparecerán en la interpretación de lo que el modelo total era. Nótese que las secuencias de costas y secuencias sonares son aquellos modelos para los cuales se envía exactamente una señal en cada intervalo de tiempo, y con la propiedad especial que aún si sólo dos señales se recuperan se puede reconstruir el modelo entero. De lo anterior se puede deducir que no hay ninguna ambigüedad en modelos. Algunos autores, describen estas propiedades como patrones que tienen una perfecta auto-correlación o auto-correlación de un hit. La meta que se quiere alcanzar en las construcciones sonares es, dado un n obtener una secuencia $m \times l$ para el mayor l posible.

El uso de las sucesiones sonares está estrechamente relacionada al codificar, envió de señales en tiempos y frecuencias determinados; para ello se requiere el disponer del mayor número secuencias posibles dadas unas condiciones iniciales particulares.

Esta investigación, está enfocada en realizar la búsqueda de sucesiones sonares computacionalmente y para ello se expone el uso de una técnica famosa (*backtracking*), para la implementación, tratando de utilizar el menor tiempo y la mayor eficiencia posible, para ello trataremos de basarnos en los resultados obtenidos mediante algunos de los modelos algebraicos para determinar algunas generalizaciones presentes en las secuencias sonares.

II. Sucesiones Sonares y Sucesiones PPM

A. *Naturaleza de las Sucesiones Sonares*

Señales sonares son usadas para determinar tanto la distancia (*rango*) de un blanco desde un observador, como la velocidad (llamada *razón de cambio de rango*) en cual el blanco se está acercando o retrocediendo del observador. El *rango* es proporcional al atraso del ciclo completo del tiempo (cambio de tiempo) de una señal, y la velocidad es proporcional al efecto "doppler" (o cambio de frecuencia) de la señal.

En un salto de frecuencia en un sistema sonar o radar, la señal consiste de una o más frecuencias existentes de un conjunto $\{f_1, f_2, \cdots, f_m\}$ de frecuencias disponibles, para la trasmisión de cada uno de un conjunto $\{t_1, t_2, \cdots, t_n\}$ de intervalos consecutivos de tiempo. Estos modelos se pueden representar convenientemente con una matriz de permutaciones A de tamaño $m \times n$, donde las m filas corresponden a las m frecuencias, las n columnas corresponden a los n intervalos de tiempo; las entradas de a_{ij} son iguales a 1 si y solo si la frecuencia f_i es trasmitida en un intervalo de tiempo t_j, en caso contrario tendremos que $a_{ij} = 0$.

Cuando las señales son reflejadas desde el objeto y son recibidas nuevamente por el observador, ellos han cambiado en ambos tiempos y frecuencias. Según la cantidad de estos cambios, es posible determinar ambos factores, distancias y velocidades. El observador determina la cantidad de estos cambios por comparación (en ambos factores, en tiempo y frecuencia) de una réplica de las señales trasmitidas con la actual señal recibida, y apuntando para cada combinación de cambio de tiempo y cambio de frecuencia la mejor coincidencia. Para llevar la cuenta de estas coincidencias se utiliza una función de correlación.

Dadas A y B dos matrices de permutación, cuando B es cambiado r unidades a la derecha (negativo si el cambio es a la izquierda), y s unidades arriba (negativo si el cambio es hacia abajo), sobreponemos la versión cambiada B^* de B en A y contamos el número de coincidencias entre las primeras en la matriz A y con las primeras en la matriz B^*. El número de estas coincidencias, $C_{AB}(r,s)$, es la función correlación entre A y B^* y claramente satisface las siguientes condiciones:

$C_{AB}(r,s) = 0$, si $|r| \geq n$ ó $|s| \geq m$, con $0 \leq C_{AB}(r,s) \leq n$

Cuando $A \neq B$, la $C_{AB}(r,s) = 0$ es llamada función de correlación cruzada. Cuando $A = B$, la $C_{AA}(r,s)$ es llamada función de auto correlación. Además de lo anterior, la función de auto correlación satisface la condición:

$$C_{AA}(0,0) = n$$

La función de auto correlación $C_{AA}(r,s)$, es llamada función ambigüedad en la literatura sonar y radar.

Supongamos que enviamos dos modelos A y B a dos blancos y entonces recibimos una matriz resonancia R (podríamos considerar algunas señales son perdidas). Al comparar R con todos los cambios de A y B, nosotros necesitamos para determinar que R es la respuesta de algún modelo A ó B, y la cantidad de ambos cambios de tiempos y frecuencias.

Si R contiene al menos tres señales y los modelos A o B satisfacen:

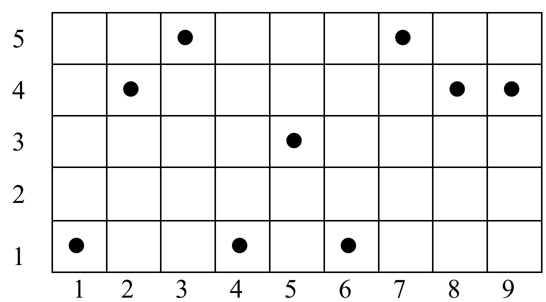
$$\begin{cases} \max C_{AB}(r,s) \leq 2 \\ \max\limits_{(r,s)\neq(0,0)} C_{AA}(r,s) \leq 2 \\ \max\limits_{(r,s)\neq(0,0)} C_{BB}(r,s) \leq 2 \end{cases}$$

También se podría construir un modelo original sin ambigüedades, para ello recurrimos a la secuencias sonares las cuales fueron sugeridas por Solomon W. Golomb y Herbet Taylor en [4] que son un modelo

$m \times n$, el cual tiene a lo más un *hit*[1] en la función de auto correlación.

En condiciones de reconocimiento de múltiples blancos, un modelo es enviado para cada blanco, las categorías de arreglos de costas y secuencias sonares para reconocimiento de múltiples objetivos son aportadas por O. Moreno y S. Maric en [9], [10] y [11]. Por el método de construcción dado en [5] se pueden construir series de secuencias las cuales pueden tener una característica de "buena correlación cruzada" pero no tener una buena característica de "buena auto correlación cruzada".

Fig. 1. Representación de una secuencia sonar de tamaño 5x9

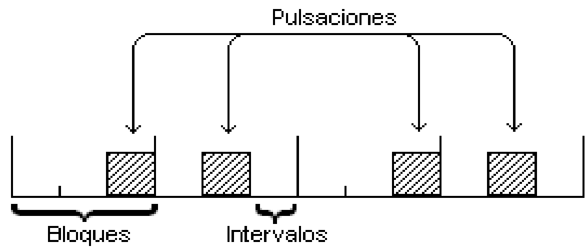


B. Origen de las Secuencias PPM

En las comunicaciones PPM (*Pulse Position Modulation*), un pulso es ubicado en uno de *m* intervalos de tiempos consecutivos para representar un *m-ésimo* símbolo. Los mensajes son enviados por la secuencia de dichos símbolos. El receptor, decodifica el mensaje generalmente mediante la integración de cada símbolo en cada intervalo y seleccionando el máximo valor integración de estos símbolos. Para lograr esta decodificación se requieren tanto el intervalo de tiempo (para fijar los intervalos de integración) como los bloques de intervalos (para enumerar correctamente los segmentos de tiempo para la identificación de estos símbolos). Lo anterior típicamente se consigue fijando intervalos de tiempo en los cuales ocurren las pulsaciones. La sincronización de estos bloques se consigue después de establecer los intervalos de tiempo y generalmente se efectúan transmitiendo algo conocido como una *secuencia PPM* (Figura 2) y usando un método de *correlación de salto* para la reconstrucción del mensaje.

Fig. 2. Representación de una secuencia PPM

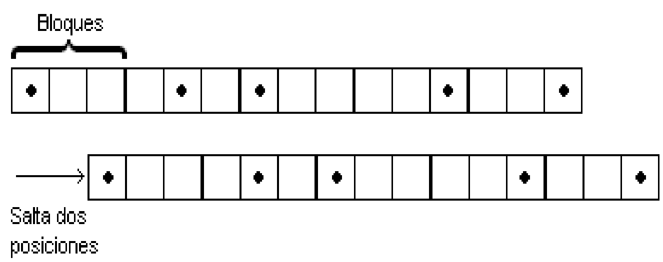
Fig. 2. Representación de una secuencia PPM



Cuando tenemos una determinada secuencia de *N* grupos de símbolos, donde cada grupo tiene *M* intervalos (o bloques) y cada intervalo cualquiera, un punto (pulsación), o un blanco (no pulsación).

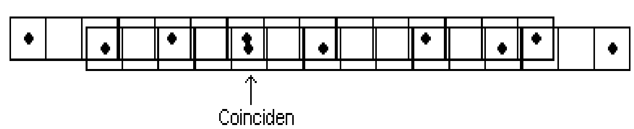
Una secuencia PPM es definida como una secuencia en la cual cada bloque contiene exactamente un punto. La correlación de salto, con una versión alternada de sí misma, es definida como el número de bloques con los cuales coinciden cuando ambas secuencias son sobrepuestas.

Fig. 3. Modelo de una secuencia ppm saltando dos posiciones



La ventaja de las secuencias PPM es que con todos los valores de correlación de salto, el máximo de coincidencias es una, es decir, la versión corrida, de todos los "saltos" posibles (diferentes de la posición original) no coinciden con el modelo original en más de un punto (figura 4).

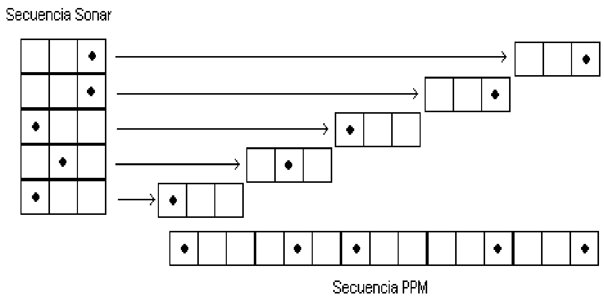
Fig. 4. Correlación de una secuencia PPM



Las secuencias PPM se podrían representar como casos particulares de arreglos de costas (cuando $N = M$) o secuencias sonares (en caso contrario). Así la representación del modelo de una PPM sería de cómo se presenta en la figura 5.

[1] Posee diferentes valores en triángulo de diferencias para dicha sucesión.

Fig. 5. Representación de una sucesión sonar mediante el modelo de una secuencia PPM



III. Uso de Algoritmos para la Búsqueda de Sucesiones Sonares y Secuencias PPM

A. Métodos Computacionales en Búsqueda de Sucesiones Sonares

La mayoría de los métodos existentes para construir arreglos de costas y sucesiones sonares son construcciones algebraicas, pero ello excluye algunas posibles arreglos que no se obtienen por uno de estos métodos algebraicos, por esta razón se han tratado de encontrar todos los posibles sucesiones, sonares o PPM, existentes utilizando métodos computacionales. Una muestra de este tipo de trabajos es el dado por O. Moreno, J. Ramírez, E. Orozco y D. Bollman [11] en el cual emplean un algoritmo utilizando *backtracking*[2] para generar un cierto número de permutaciones. Luego de trabajar este tipo de generación de permutaciones se estima que utilizando la ayuda de un algoritmo con *backtracking* se pueden generar arreglos de costas pero que ello requiere cierto costo computacional.

"*Backtracking*" (también conocido como "vuelta atrás") es una técnica que de forma sistemática y organizada, genera y recorre un espacio que contiene todas las posibles secuencias de decisiones. A este espacio se le denomina "espacio de búsqueda del problema".

Los algoritmos de *backtracking* hacen una búsqueda sistemática de todas las posibilidades, sin dejar ninguna por considerar. Cuando intenta una solución que no lleva a ningún sitio, retrocede deshaciendo el último paso, e intentando una nueva variante desde esa posición (es normalmente de naturaleza recursiva).

La técnica de *backtracking* ofrece un método para resolver problemas tratando de completar una o varias soluciones por etapas [4]. En cada paso se intenta extender una solución parcial de todos los modos posibles, y si ninguno resulta satisfactorio se produce un retroceso (o vuelta atrás) hasta el último punto donde quedaban alternativas por explorar. Es una mejora de la búsqueda completa de soluciones y una alternativa a los algoritmos voraces.

En muchas aplicaciones del método de *backtracking*, la solución deseada se puede expresar como una n-tupla $(x_1, x_2, \ldots, x_n)$, donde cada x_i hace parte de la selección de un conjunto finito S_i. Frecuentemente para resolver el problema (encontrar un vector máximo, o mínimo o que satisfaga las condiciones deseadas) se utiliza una *"función criterio"* $P(x_1, x_2, \ldots, x_n)$.

En nuestro caso particular queremos encontrar todas las posibles soluciones factibles (secuencias sonares dados m y n), por eso es bien conveniente utilizar esta técnica. Para una sucesión $s = \{s_1, s_2, \ldots, s_n\}$, tendremos m^n soluciones posibles. La función "prueba", va a hacer una selección de las sucesiones las cuales satisfacen la condición del "triángulo de diferencias". La restricción de este problema va a ser entonces dada en poder verificar la prueba del triángulo de diferencias.

B. El Algoritmo

Sea *m* el número máximo de cada valor de la sucesión y sea *n* el largo de dicha sucesión, el proceso utilizado en el algoritmo para determinar las sucesiones sonares es el siguiente.

Utilizando el algoritmo de "Backtracking" se inicia a hacer la búsqueda de las sucesiones factibles que cumplen con la condición de las "diferencias distintas".

Inicialmente se empieza por los valores más pequeños de la sucesión y se van aumentando términos a medida que la condición se cumple. Si se encuentra una "sub-sucesión" que no cumple la condición, se aborta el proceso. O sea, dada una sucesión $S = \{\ s_1\ s_2\ \cdots\ s_n\ \}$, si $s_1\ s_2\ \cdots\ s_k$ no es secuencia sonar, entonces, $s_1\ s_2\ \cdots\ s_k\ s_{k+1}$ tampoco será secuencia sonar. (Siendo $k < n$).

La forma de determinar si una sucesión es una sucesión factible es la siguiente (para ilustrarlo de una mejor forma vamos a tomar unos valores fijos de $n = 5$ y $m = 3$):

[2] Técnica de búsqueda en secuencias numéricas

Inicialmente se describe la llamada matriz de diferencias, una matriz que posee $n-1$ filas (niveles de diferencias) y $2\times(m-1)$ columnas (posibles diferencias). Esta matriz inicialmente está llena de "ceros", para nuestro ejemplo en particular tendremos la siguiente matriz de tamaño 5×4

$$\begin{array}{c|ccccc} & -2 & -1 & 0 & 1 & 2 \\ \hline n1 & 0 & 0 & 0 & 0 & 0 \\ n2 & 0 & 0 & 0 & 0 & 0 \\ n3 & 0 & 0 & 0 & 0 & 0 \\ n4 & 0 & 0 & 0 & 0 & 0 \end{array}$$

Partimos con el valor base: { 1 }, añadimos un término más a la sucesión { 1 1 }, encontramos la respectiva diferencia se verifica que esta diferencia no ha sido seleccionada anteriormente (es decir que esté cero su valor) y se "registra" (modificando el valor de cero por 1) en la matriz de diferencias de la siguiente forma:

$$\begin{array}{ccccc} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{array}$$

Luego agregamos el siguiente valor en la sucesión: { 1 1 1 }, luego calculamos las respectivas diferencias por niveles en orden inverso (el último menos el penúltimo en el nivel 1 y el ultimo menos el antepenúltimo en el nivel dos), así al iniciar a hacer la primera diferencia observamos que el su valor es 0, pero al hacer el registro notamos que ya este valor había sido tomado y por tal razón no va a funcionar. En este momento se aborta el cálculo de diferencias para pasar a buscar con el siguiente valor en la sucesión es decir: { 1 1 2 }, hacemos las diferencias y vamos registrando en la matriz. 2 – 1 = 1 para el nivel 1 y 2 – 1 = 1 para el nivel 2. en este caso la matriz de diferencias será:

$$\begin{array}{ccccc} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{array}$$

Como las condiciones se cumplieron, verificamos que como aún no ha llegado a la máxima longitud de la sucesión, procedemos a agregar un término más a la sucesión { 1 1 2 1 }, hacemos la verificación de las diferencias, n1: 1 – 2 = – 1, n2 : 1 – 1 = 0 y n3: 1 – 1 = 0 . Así la matriz de diferencia en este momento será:

$$\begin{array}{ccccc} 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{array}$$

Ahora, como se satisface la propiedad pasamos a agregar un valor más en la sucesión: { 1 1 2 1 1 }. Verificamos las diferencias: n1: 1 – 1 = 0, como esta ocupado, abortamos y cambiamos a { 1 1 2 1 2 }. Las diferencias allí serán: n1 = 2 – 1 = 1, como está lleno nuevamente abortamos el cálculo de las diferencias y cambiamos la sucesión a: { 1 1 2 1 3 }, de donde las diferencias serán: n1: 3 – 1 = 2, n2 : 3 – 2 = 1, como n2 ya estaba lleno con 1 se aborta el proceso y se pasa al siguiente componente el cual sería 4 pero este valor es mayor de *m*. Ahora debemos hacer "back", para ello debemos borrar las diferencias que ha puesto los valores que vamos a retroceder. Así vamos a tener que ir hacia atrás a { 1 1 2 1 } y pasando a { 1 1 2 2 }. Como retrocedimos e 1 de la cuarta posición, también debemos eliminar las diferencias que este valor había dejado y la matriz de diferencias será la siguiente:

$$\begin{array}{ccccc} 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{array}$$

En el siguiente paso, se calculan las diferencias para { 1 1 2 2 }. Las diferencias serán: n1: 2 – 2 = 0, abortamos, y pasamos a { 1 1 2 3 }, hacemos las diferencias : n1: 3 – 2 = 1, abortamos nuevamente tenemos que hacer nuevamente un "back" para llegar a { 1 1 3 } y por tal razón deberíamos borrar las diferencias registradas por el 2 de la tercera posición. Así la matriz de diferencias quedará en:

$$\begin{array}{ccccc} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{array}$$

Para que registrar las diferencias puestas por el 3 de la tercera posición en n1: 3 – 1 = 2 y n2: 3 – 1 = 2, de la siguiente forma:

$$\begin{matrix} 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$$

Nuevamente vamos hacia adelante con { 1 1 3 1 }, n1: 1 – 3 = –2, n2: 1 – 1 = 0, y n3: 1 – 1 = 0. La matriz de diferencias será:

$$\begin{matrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$$

Pasando a la sucesión { 1 1 3 1 2 }, y registramos sus diferencias, n1: 2 – 1 = 1, n2 : 2 – 3 = –1, n3: 2 – 1 = 1 y n4: 2 – 1 = 1. Así tenemos:

$$\begin{matrix} 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \end{matrix}$$

Una vez alcanzado el valor máximo de *n* se procede a imprimir una sucesión factible, para pasar a la siguiente sucesión factible primero debemos retirar los valores de las diferencias de la última componente y verificar si cumplen lo de la siguiente, obteniendo la siguiente matriz:

$$\begin{matrix} 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{matrix}$$

Se Agrega un nuevo componente { 1 1 3 1 3 }, hacemos las diferencias, n1: 3 – 1 = 2, como está lleno abortamos y vamos hacia atrás.

Este proceso se sigue hasta haber hecho las búsquedas de todas las sucesiones posibles.

IV. Conclusiones

La búsqueda de sucesiones sonares mediante el uso de medios computacionales ofrece la posibilidad de encontrar todas las sucesiones existentes para las condiciones iniciales dadas y con esta ayuda se puede favorecer ideas otros métodos de construcciones de sucesiones algebraicos.

Dadas las condiciones del problema propuesto, definitivamente la técnica de “Backtracking” es la que más favorece en la búsqueda exhaustiva de sucesiones, puesto que además de hacer una búsqueda ordenada y dirigida, controla los campos de búsqueda, a la vez que descarta las posibilidades donde no va a encontrar soluciones factibles.

El algoritmo de búsqueda de secuencias arroja soluciones que pudieron ser verificadas con tablas de resultados algebraicos y se pueden corroborar con las soluciones exactas.

El tiempo de ejecución es bastante bajo para la configuración del hardware donde se realizaron las pruebas Esto nos lleva a inferir que, con una configuración más potente sí se podría pasar esta barrera.

Referencias

[1] G. Costas, “On PPM Sequences whit Good Autocorrelation Properties,” IEEE Trans. Inform. Theory, vol. 35, NO. 1, pp. 146-149, Mayo 1988.

[2] R. Gagliardi, J. Robbins, y H. Taylor, “Acquisition sequences in PPM communications,” IEEE Trans. Inform. Theory, vol. IT-33, pp. 738-744, 1987.

[3] R. A Games, “An algebraic construction of sonar sequences using M-sequences,” SIAM J. Algebraic Discrete Methods, vol. 8, pp. 753 – 761, Octubre 1987.

[4] S. W. Golomb,L. Baumert, “Backtrack Programming,” Journal of the ACM (JACM), vol. 12, pp. 516-524, 1965.

[5] S. W. Golomb, “Algebraic constructions for Costas arrays,” J. Combinatorial Theory, Ser. A, vol. 37, pp. 13-21, 1984.

[6] S. W. Golomb y H. Taylor, “Two-dimensional

synchronization patterns for minimum ambiguity," IEEE Trans. Inform. Theory, vol. IT-28, pp. 263-272, Julio 1982.

[7] S. W. Golomb y H. Taylor, "Constructions and properties of Costas arrays," Proc. IEEE, vol. 72, pp. 1143-1163, Septiembre 1984.

[8] S.V. Maric y E. L. Titlebaum, "A class of Frequency Hop Codes with Nearly Ideal Characteristics for Use in Multiple Access Spread Spectrum Communications and Radar and Sonar System," en IEEE Transactions on Communications, Septiembre 1992

[9] O. Moreno, S Maric "Classes of Costas and Sonar Sequences For Multi-target Recognition," en IEEE ISIT 1997.

[10] O. Moreno, S Maric and LiYuchun "Best Know Sonar Sequences for Multi-target Recognition. In IEEE ISIT 1997.

[11] O. Moreno y R. A. Games, "Sonar Sequences from Costas Arrays and the Best Known Sonar Sequences with up to 100 Symbols," IEEE Trans. Inform. Theory, vol. 39, NO. 6, pp. 1985-1987, Noviembre 1993.

[12] O. Moreno. S.V. Maric, "A Class of Frequency Hop Codes with Nearly Ideal Characteristics for Multiple-target Recognition. In thirty-third Annual Allerton Conference on Communication, Control, and Computing, Octubre 1995.

[13] O. Moreno, P. Pei and J. Ramirez, A Parallel Algorith for Enumeration of Costas, In proceedings of the 7th SIAM Conference on Paralllel Conference Processing for Scientific Computing pp. 255-260, 1995

[14] O.Moreno, J. Ramirez, D. Bollman, y E. Orozco, "Faster algorithms for the generation of Symmetry-Invariant Permutations", Marzo de 2002.

[15] E. Orozco, "On the parallel generation of coastal arrays," Tesis, University of Puerto Rico,1998.