

# Virtual bat algorithm for the computation of Duhamel's Integral applied to structural systems with one degree of freedom

## Algoritmo del murciélago virtual en el desarrollo de la Integral de Duhamel para sistemas estructurales con un grado de libertad

Sergio Vanegas\*, Iván Amaya<sup>1\*\*</sup>, Rodrigo Correa<sup>\*\*\*</sup>

\* Estudiante de Maestría en Ingeniería Civil, Escuela de Ingeniería Civil. Universidad Industrial de Santander, Bucaramanga. COLOMBIA

\*\* Estudiante de Doctorado en Ingeniería, Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones. Universidad Industrial de Santander, Bucaramanga. COLOMBIA

\*\*\* Profesor Titular, Escuela de Ingeniería Ingenierías Eléctrica, Electrónica y de Telecomunicaciones. Universidad Industrial de Santander, Bucaramanga. COLOMBIA

Fecha de Recepción: 12/12/2012

Fecha de Aceptación: 06/06/2013

PAG 278 - 289

### Abstract

The current article presents the use of the virtual bat optimization algorithm, for solving Duhamel's Integral in dynamic systems with one degree of freedom. Some situations are solved, as a demonstrative example, calculating global maxima. The results found were comparable (both, in precision and in accuracy) to those achieved with other calculation approaches. In a similar fashion, accelerometer data of a Chilean seism (2010) was used to calculate the dynamic response and its maximum displacement. In all cases, the number of virtual bats were varied, noting an influence over the run time.

Keywords: Bat algorithm, Duhamel's Integral, dynamic structure, impulsive excitation, earthquake accelerograms

### Resumen

En este artículo se propone el uso del algoritmo de optimización de murciélagos virtuales para resolver la Integral de Duhamel en sistemas dinámicos con un grado de libertad. Se desarrollaron ejercicios a título demostrativo, para el cálculo de máximos globales. Los resultados fueron muy comparables, tanto en precisión, como en exactitud frente a aquellos obtenidos por otros métodos de cálculo. Se utilizó igualmente el acelerograma del sismo de Chile (2010), para el que se reporta la respuesta dinámica y el desplazamiento máximo. Se variaron para estos ejemplos la cantidad de murciélagos virtuales notándose su influencia sobre el tiempo de computación.

Palabras Clave: Algoritmo de murciélagos, Integral de Duhamel, dinámica estructural, excitaciones impulsivas, acelerograma de sismos

## 1. Introducción

Al realizar el análisis de dinámica estructural en estructuras de obras civiles, a menudo se descubre que éstas se encuentran sujetas a excitaciones arbitrarias, las cuales, por lo general, no son armónicas. La respuesta dinámica a estas excitaciones (que pueden deberse a excitaciones por impulsos o a la actividad sísmica del terreno), pueden ser predichas a través del cálculo de una integral que puede ser analíticamente evaluada, cuando la señal de excitación corresponde a una simple expresión. Esta integral se conoce como la Integral de Duhamel (ID) que permite, en teoría, calcular la respuesta de un sistema lineal (estructura arbitraria), con un único grado de libertad, debido a excitaciones por impulsos. Esta integral, por definición, es aplicada durante un intervalo corto de tiempo. El impulso, que le corresponde, puede ser definido como el producto de la fuerza por el tiempo de duración.

Un sismo, que también puede ser considerado como una excitación por impulsos, es analizado a través de la integral antes mencionada. En este caso se requiere contar con la información del movimiento telúrico, proporcionada por un acelerograma, que consiste básicamente en series de datos (aceleraciones) separados por intervalos breves de tiempo.

## 1. Introduction

When performing a dynamic analysis of civil structures, it is often found that they are subject to arbitrary excitations, which, in a general case, are not harmonic. The dynamic answer to these excitations (that can be provoked by impulsive excitations or by accelerations of the ground due to seismic activity), can be predicted through the calculation of an integral that can be analytically evaluated when the excitation signal is a simple expression. This integral is known as Duhamel's Integral (DI), and it allows to theoretically calculate the response of a linear system (arbitrary structure) with a single degree of freedom and due to impulsive excitations. The latter, by definition, is an excitation applied for a short time interval. The impulse belonging to it can be defined as the product of a force by its duration (time).

A seism, which can also be considered an impulsive excitation, can be analyzed through the aforementioned integral. In this case, information of the telluric movement is required, which is gathered through an accelerogram, and which basically is a data series (accelerations) separated by short time intervals.

<sup>1</sup> Autor de correspondencia / Corresponding author:  
E-mail: ivan.amaya2@correo.his.edu.co

Este artículo muestra, a modo de ejemplo, que es posible analizar (evaluar) la integral de Duhamel a través de una estrategia de optimización numérica, conocida como Algoritmo de Murciélagos, para obtener los desplazamientos máximos y mínimos de la respuesta dinámica de un sistema, con un único grado de libertad.

Los algoritmos de optimización meta-heurísticos se han hecho populares, debido a su simplicidad y facilidad de programación. Más aún, proporcionan buenos resultados (precisión y exactitud) al optimizar problemas con un relativo grado de complejidad, y pueden operar en sistemas con restricciones. Entre ellos, podemos citar la Optimización por Enjambre de Partículas (PSO) (Parsopoulos & Vrahatis 2010), así como una de sus más recientes modificaciones, es decir Optimización por Enjambre de Partículas Binarias (BPSO) (Kennedy & Eberhart 1997), Búsqueda de Armonía (Ramos et al., 2011), y el Algoritmo de Enjambre (Xing-She Yang 2010), son algunos de los más destacados. Existe otro planteamiento importante, es decir el algoritmo del murciélago (AM), que fue propuesto en el año 2010. Éste imita el comportamiento de micro murciélagos en búsqueda de una presa, con una impresionante habilidad para diferenciar entre obstáculos y sus objetivos. El presente artículo trata sobre el uso de este algoritmo para resolver la Integral de Duhamel, y se enfoca en el efecto producido al variar el tamaño de la población, sobre la capacidad de encontrar una respuesta apropiada.

## 2. Discusión y desarrollo

### 2.1 Fundamento teórico

La Integral de Duhamel es una ecuación teórica que permite el cálculo de la respuesta de un sistema lineal, con un grado único de libertad, para estructuras arbitrarias, que se encuentran bajo una excitación breve y externa. En el caso generalizado de un sistema amortiguado, la ID tiene la siguiente forma (Paz 1992; Maldonado & Chio 2004):

$$u(t) = \frac{1}{m\omega_D} \int_0^t P(\tau) e^{-\xi\omega(t-\tau)} \text{sen} [\omega_D(t-\tau)] d\tau \quad (1a)$$

Donde/ where:

$m$  = mass of the system (Masa del sistema)

$\omega_D$  = damped angular frequency =  $\omega\sqrt{1-\xi^2}$  (Frecuencia angular amortiguada)

$\omega$  = natural frequency of the system =  $\sqrt{k/m}$  (Frecuencia natural del sistema)

$k$  = system stiffness (Rigidez del sistema)

$\xi$  = damping coefficient (Coeficiente de amortiguación)

$P(\tau)$  = external force at instant  $\tau$  (function that simulates the external impulse) (Fuerza externa en el momento  $\tau$  – función que simula el impulso externo)

$u(t)$  = time displacement (Tiempo de desplazamiento)

*This article shows, in an exemplary way, how it is possible to analyze (evaluate) Duhamel's Integral through a numeric optimization strategy known as Virtual Bat Algorithm (VBA), to obtain the maximum and minimum displacements of the dynamic response of a system with a single degree of freedom.*

*Metaheuristic optimization algorithms have become popular because of their simplicity and easiness of programming. Moreover, they provide good results (i.e. precision and accuracy) in optimization problems with a relative degree of complexity, and they can operate on systems with restrictions. Among them, Particle Swarm Optimization (PSO) (Parsopoulos & Vrahatis 2010), as well as one of its more recent modifications, i.e. Binary Particle Swarm Optimization (BPSO) (Kennedy & Eberhart 1997), Harmony Search (HS) (Ramos et al., 2011), and the Firefly algorithm (Xing-She Yang 2010), are some of the more outstanding ones. There is also another important approach, i.e. the bat algorithm (BA), which was proposed in 2010. This mimics the behavior of microbats when looking for prey, with the impressive ability of being able to differentiate between obstacles and their target. The current article deals with the use of this algorithm for solving Duhamel's Integral, and focuses on the effect of varying the population size on the capability of finding an appropriate answer.*

## 2. Discussion and development

### 2.1 Theoretical foundation

*Duhamel's Integral is a theoretical equation that allows calculation of the response of a linear system with a single degree of freedom, for arbitrary structures, which are under an external and brief excitation. For the general case of a damped system, DI has the following form (Paz 1992; Maldonado & Chio 2004):*

Al tratar con un sistema sin amortiguación ( $\xi=0$ ), ID se transforma en:

When dealing with a system with no damping ( $\xi=0$ ), DI becomes:

$$u(t) = \frac{1}{m\omega_n} \int_0^t P(\tau) \text{sen} [\omega_D (t - \tau)] d\tau \quad (1b)$$

Y luego de un cambio trigonométrico, da como resultado:

And after a trigonometric change, it yields:

$$u(t) = [A(t) \text{sen} \omega_D t - B(t) \text{cos} \omega_D t] \frac{e^{-\xi\omega t}}{m\omega_n} \quad (1c)$$

Siendo/Being:

$$\begin{aligned} A(t) &= \int_0^t P(\tau) e^{\xi\omega\tau} \text{cos} \omega_D \tau d\tau \\ B(t) &= \int_0^t P(\tau) e^{\xi\omega\tau} \text{sen} \omega_D \tau d\tau \end{aligned} \quad (1d)$$

### 2.1.1 Respuesta espectral por excitación en el soporte (movimiento sísmico)

Un importante problema, desde el punto de vista de la estructura, es el análisis de un sistema bajo el efecto de excitación en la base de la estructura (fundación o soporte). La ecuación de movimiento del oscilador amortiguado, bajo excitación, es (Maldonado & Chio 2004):

### 2.1.1 Spectral response for an excitation at the support (seismic movement)

An important problem, from the point of view of the structure, is the analysis of a system under the effect of an excitation at the base (foundation or support) of the structure. The movement equation of the damped oscillator under excitation is (Maldonado & Chio 2004):

$$m\ddot{u} + c(\dot{u} - \dot{u}_s) + k(u - u_s) = 0 \quad (1e)$$

Donde/for which:

$m$  = mass of the system (Masa del sistema)

$\ddot{u}$  = acceleration of the system (Aceleración del sistema)

$\dot{u}$  = speed of the system (Velocidad del sistema)

$u$  = displacement of the system at the highest point of the structure (Desplazamiento del sistema en el punto más alto de la estructura)

$u_s$  = displacement of the system at its base (Desplazamiento del sistema en su base)

Esta ecuación también puede ser escrita en función del movimiento relativo de la masa, con respecto al movimiento del soporte, obteniéndose:

This equation can also be written as a function of the relative movement of the mass with respect to the movement of the support, obtaining:

$$y + 2\xi\omega\dot{y} + \omega^2 y = -\ddot{u}_s(t) \quad (1f)$$

Donde  $y$  es el movimiento relativo, que puede ser definido como:

where  $y$  is the relative movement, that can be defined as:

$$y = u - u_s \quad (1g)$$

La solución de esta ecuación, en respuesta a la deformación  $\mathbf{y}(\mathbf{t})$ , básicamente depende de la aceleración del movimiento del terreno  $\mathbf{u}_s''(\mathbf{t})$ , de la frecuencia natural de vibración  $\omega = \sqrt{\mathbf{K}/\mathbf{m}}$ , y del índice de amortiguación,  $\xi$ , de la estructura. La solución analítica de esta ecuación diferencial es:

$$\mathbf{y}(\mathbf{t}) = -\frac{1}{\omega_n} \int_0^{\mathbf{t}} \ddot{\mathbf{u}}_s(\tau) e^{-\xi\omega(\mathbf{t}-\tau)} \text{sen} [\omega_D(\mathbf{t}-\tau)] d\tau \quad (1h)$$

Puesto que la aceleración sísmica del terreno varía, en forma irregular en el tiempo, es imposible encontrar una solución analítica a este problema. Sin embargo, existen procedimientos numéricos, basados en el constante aumento de la velocidad de procesamiento de los computadores modernos, que son una alternativa efectiva para ocuparse de tales situaciones. Los acelerogramas sísmicos son actualmente digitalizados y filtrados, para controlar y depurar errores iniciales, y así obtener un acelerograma apropiado del terreno. La función  $\mathbf{u}_s''(\mathbf{t})$  representa el acelerograma definido por las coordenadas numéricas, corregidas en intervalos de tiempo bastante ajustados, como para proporcionar una aproximación adecuada.

Una vez resuelta la ecuación de movimiento, se obtiene una respuesta del tiempo del acelerograma, en toda su extensión. Con lo anterior es posible definir la respuesta máxima alcanzada por el sistema, que representa una variable clave para predecir su comportamiento cuando enfrenta un sismo de similares características.

### 2.1.2 Algoritmo del Murciélago Virtual (AMV)

Este artículo propone la transformación del cálculo del desplazamiento máximo con la ID, en un problema de maximización. Más aún, se decidió emplear el AMV para este propósito. Este algoritmo fue propuesto inicialmente por Yang, y está basado en la descripción aproximada de algunas especies de murciélagos, por sus características naturales de eco-localización, que puede ser empleada para ubicar objetos y su presa, a través del empleo del eco producido por el zumbido ultrasónico, que rebota entre los objetos rodeados. Este sistema es tan preciso que puede evitar obstáculos tan pequeños como el cabello humano. Se cree que también se puede generar un escenario 3D empleando esta técnica e incluyendo la diferencia de tiempo, entre la percepción en sus oídos, y la variación de intensidad de los ecos (X. Yang 2010).

A pesar de haber sido propuesto en el año 2010, existe evidencia de algunas variantes que se esmeran por mejorar su comportamiento, como el planteamiento de optimización (Xing-She Yang 2010; X.-S. Yang & Gandomi 2012). Con el fin de simplificar el algoritmo, se establecieron algunas reglas básicas en relación al comportamiento de los murciélagos (X. Yang 2010; Gandomi et al., 2012; X. Yang 2011):

1. Los murciélagos virtuales emplean eco-localización para diferenciar entre obstáculos y sus presas.
2. Los murciélagos virtuales vuelan aleatoriamente, a una velocidad  $v_i$ , en una posición  $x_i$ , emitiendo una frecuencia entre  $f_{min}$  y  $f_{max}$ , variando la longitud de onda  $\lambda$  y la intensidad  $A_0$  cuando localizan a su presa.

*The solution of this equation to the response of the deformation  $\mathbf{y}(\mathbf{t})$ , depends, basically, on the acceleration of ground's movement,  $\mathbf{u}_s''(\mathbf{t})$ , the natural vibrating frequency  $\omega = \sqrt{\mathbf{K}/\mathbf{m}}$ , and the damping ratio,  $\xi$ , of the structure. The analytical solution of this differential equation is:*

*Due to the fact the seismic acceleration of the terrain varies irregularly with time, an analytical solution of the problem is just impossible. However, numerical procedures, supported on the constantly increasing processing velocity of modern computers, pose an effective alternative for dealing with these situations. Seismic accelerograms are already digitalized and filtered to control and debug initial errors, thus obtaining an appropriate accelerogram of the terrain. The function represents  $\mathbf{u}_s''(\mathbf{t})$  the accelerogram defined by the corrected numerical coordinates in some time intervals close enough to provide an appropriate approximation.*

*Having solved the movement equation, a response throughout the time of the accelerogram, is obtained. With it, it is possible to define the maximum response achieved by the system, which represents an important variable for predicting its behavior when facing a seism with similar characteristics.*

### 2.1.2 Virtual bat algorithm (VBA)

*During this article it is proposed to transform the calculation of the maximum displacement through DI, into a maximization problem. Moreover, it was decided to use BA for this task. This algorithm was initially proposed by Yang, and it is based on the approximate description of some bat species given their natural echolocation features, which can be used to locate objects and prey using the echo produced by their ultrasonic squeal, that bounces between the surrounding objects. This system is so developed that they can avoid obstacles as small as a human hair. It is believed that they also generate a 3D scenario using this technique, as well as the time difference between the perception at their ears, and the variation in the intensity of the echoes (X. Yang 2010).*

*In spite of being proposed in 2010, there is already evidence of some variants that strive to improve its behavior as an optimization approach (Xing-She Yang 2010; X.-S. Yang & Gandomi 2012). In order to simplify the algorithm, some ground rules were established regarding the behavior of bats (X. Yang 2010; Gandomi et al., 2012; X. Yang 2011):*

1. *Virtual bats use echolocation to differentiate between obstacles and prey.*
2. *Virtual bats fly randomly with velocity  $v_i$  at a position  $x_i$ , emitting a frequency between  $f_{min}$  and  $f_{max}$ , varying the wavelength  $\lambda$  and the intensity  $A_0$  when locating their prey.*

Pueden ajustar la longitud de onda (o la frecuencia) y la emisión de la relación  $r$ ,  $r \in [0,1]$  dependiendo de la proximidad con su objetivo.  
3. La intensidad varía de una velocidad inicial  $A_0$  a un mínimo de  $A_{min}$ .

En resumen, el pseudo-código del algoritmo puede ser expresado tal como lo propuso Yang, (X. Yang 2010; Gandomi et al., 2012; X. Yang 2011; Nakamura et al., 2012; Ponnappa & Pupong 2012):

Defina la función objetivo  $f(x)$ ,  $x = (x_p, \dots, x_d)^T$ .  
Inicialice la población virtual de murciélagos  $x_i$  ( $i = 1, 2, \dots, n$ ) y  $v_i$   
Defina las frecuencias  $f_i$  at  $x_i$   
Inicialice la relación  $r_i$  y su intensidad  $A_i$ .  
**Mientras** ( $t < \text{Máx. Número de iteraciones}$ )  
Genere nuevas soluciones ajustando las frecuencias, y actualizando la velocidad y posición de cada murciélago, empleando las ecuaciones (1) a (3).  
**Si** (*aleatorio*  $> r_i$ )  
Seleccione una solución dentro de las mejores.  
Genere una solución local sobre la mejor solución seleccionada, empleando un recorrido aleatorio.  
**Finalizar si**  
**Si** (*aleatorio*  $A_i \& f(x_j) < f(x_*)$ )  
Acepte las nuevas soluciones  
Aumente  $r_i$  y reduzca  $A_i$ .  
**Finalizar si**  
Clasifique los murciélagos y encuentre el mejor  $x_*$  real.  
**Finalizar mientras**

Durante las simulaciones, la inicialización de las frecuencias, así como la posición y velocidad de cada murciélago, son definidas empleando las Expresiones (2) a la (4):

$$f_i = f_{min} + (f_{max} - f_{min})\beta \quad (2)$$

$$v_i^t = v_i^{t-1} + (x_i^t - x_*)f_i \quad (3)$$

$$x_i^t = x_i^{t-1} + v_i^t \quad (4)$$

Donde/ where:

$\beta$ = Valor aleatorio entre 0 y 1/Random value between 0 and 1.

$f_i$ =Frecuencia para el murciélago virtual  $i$ -ésimo/ Frequency for the  $i$ -th virtual bat.

$f_{min}, f_{max}$ =Frecuencias mínimas y máximas. Valores recomendados entre 0 y 100 (X. Yang 2010)/Minimum and maximum frequencies. Values between 0 and 100 are recommended (X. Yang 2010).

$v_i^t, v_i^{t-1}$ =Velocidad en el paso de tiempo  $t$  y  $t-1$ , respectivamente, para el murciélago  $i$ -ésimo/ Velocity at time step  $t$  and  $t-1$ , respectively, for the  $i$ -th bat.

$x_i^t, x_i^{t-1}$ =Posición en el paso de tiempo  $t$  y  $t-1$ , respectivamente, para el murciélago  $i$ -ésimo/ Position at time step  $t$  and  $t-1$ , respectively, for the  $i$ -th bat.

$x_*$ =Mejor posición en el actual paso de tiempo/ Best position at the current time step.

They can adjust the wavelength (or the frequency) as well as the emission ratio  $r$ ,  $r \in [0,1]$  depending on the proximity to their target.

3. The intensity varies from an initial value  $A_0$  to a minimum of  $A_{min}$ .

Summarizing, a pseudocode of the algorithm can be expressed as the original proposal of Yang (X. Yang 2010; Gandomi et al., 2012; X. Yang 2011; Nakamura et al., 2012; Ponnappa & Pupong 2012):

Define the objective function  $f(x)$ ,  $x = (x_p, \dots, x_d)^T$ .  
Initialize the virtual bats population  $x_i$  ( $i = 1, 2, \dots, n$ ) and  $v_i$   
Define frequencies  $f_i$  at  $x_i$ .  
Initialize the pulse rate  $r_i$  and their intensity  $A_i$ .  
**while** ( $t < \text{Max. iteration number}$ )  
Generate new solutions by adjusting the frequencies, and update each bat's velocity and position, using equation. (1) to equation (3).  
**if** (*rand*  $> r_i$ )  
Select a solution between the best ones.  
Generate a local solution around the best solution selected, using a random walk.  
**end if**  
**if** (*rand*  $< A_i \& f(x_j) < f(x_*)$ )  
Accept the new solutions.  
Increase  $r_i$  and reduce  $A_i$ .  
**end if**  
Classify bats and find the actual best  $x_*$ .  
**end while**

During simulations, the initialization of the frequencies, as well as the position and velocity of each bat, is defined using Expressions (2) to (4):

La búsqueda local de este algoritmo es ejecutada empleando un recorrido aleatorio, proporcionado por la Ecuación (5):

*The local search in the algorithm is carried out using a random walk, that is given in Equation (5):*

$$x_{new} = x_{old} + \epsilon \bar{A}^t \quad (5)$$

Donde/ where:

$\epsilon$  = Número aleatorio entre -1 y 1 / Random number between -1 and 1.

$\bar{A}^t$  = Promedio de  $A_i$  en el actual paso de tiempo / Average of  $A_i$  at the current time step

Para aumentar  $r_i$  y  $A_i$ , se emplean las Expresiones (6) y (7):

*In order to increase  $r_i$  and  $A_i$ , Expressions (6) and (7) are used:*

$$A_i^{t+1} = \alpha A_i^t \quad (6)$$

$$r_i^{t+1} = r_i^0 [1 - \exp(-\gamma t)] \quad (7)$$

Donde/ where:

$\alpha$  = Valor aleatorio entre 0 y 1. En esta investigación, se escogió el valor 0.9 (X. Yang 2010) / Random value between 0 and 1. During this research, a value of 0.9 was chosen (X. Yang 2010).

$\gamma$  = Valor constante mayor que cero. Esta investigación asumió el valor 0.9 (X. Yang 2010) / Constant value bigger than zero. During this research, 0.9 was assumed (X. Yang 2010).

### 2.1.3 Murciélago Virtual para la Integral de Duhamel

Con el fin de implementar el AMV para resolver la ID, fue necesario reemplazar la demo entregada por Matlab Central n.d., por la siguiente:

*Función objetivo (Expresión (1))*

*Inicialice la población de murciélagos virtuales  $x_i$  ( $i = 1, 2, \dots, n$ ) y  $v_i$  a en un dominio determinado (definido entre 0 y t-max).*

*Defina las frecuencias  $f_i$  en  $x_i$  (definidas entre  $f_{min}=0$  y  $f_{max}=2$ ).*

*Inicialice pulsando relación  $r_i=0.5$  y su intensidad  $A_i=0.5$  (constante para cada paso de tiempo).*

*Defina un error  $E = \infty$ .*

*Defina una tolerancia para criterio de finalización  $Tol = 0.01$*

**Mientras** ( $t < \text{Máximo número de iteraciones} \ \&\& \ E > Tol$ )

*Evalúe la función (Ecuación (1)) y calcule  $f(x'_i)$  (en este caso,  $u(t)$ ).*

*Busque el valor máximo o mínimo ( $x_*$ ), entre los valores previamente evaluados.*

*Genere nuevas soluciones, ajustando frecuencias y actualizando las velocidades y posición [Expresiones (1) a la (3)]. Si las nuevas soluciones,  $x'_p$ , se encuentran fuera de los límites, vuelva atrás.*

*Si (aleatorio  $> r$ )*

*Genere una nueva solución local sobre la mejor solución seleccionada,  $x_*$ , a través de un recorrido aleatorio:*

### 2.1.3 Virtual bat for Duhamel's Integral

*In order to implement VBA for solving DI, it was necessary to modify the demo provided by (Matlab Central n.d.), with the following one:*

*Objective Function (Expression (1))*

*Initialize virtual bats population  $x_i$  ( $i = 1, 2, \dots, n$ ) and  $v_i$  a given domain (defined between 0 and t-max).*

*Define frequencies  $f_i$  at  $x_i$  (defined between  $f_{min}=0$  and  $f_{max}=2$ ).*

*Initialize pulse rate  $r_i=0.5$  and their intensity  $A_i=0.5$  (constants for each time step).*

*Define an error  $E = \infty$ .*

*Define a tolerance for stop criteria,  $Tol=0.01$*

**while** ( $t < \text{Max iteration number} \ \&\& \ E > Tol$ )

*Evaluate the function (Equation (1)) and calculate  $f(x'_i)$  (for this case,  $u(t)$ ).*

*Look for the maximum or minimum ( $x_*$ ) on the previously evaluated values.*

*Generate new solutions by adjusting frequencies, and update velocities and position [Expressions (1) to (3)]. If the new solutions,  $x'_p$ , are out of bounds, then bring them back.*

*if (rand  $> r_p$ )*

*Generate a new local solution around the best solution selected,  $x_*$ , through a random walk:*

$$x_{new} = x_* + 0.01 * rand(0, 1) \quad (8)$$

**Finalizar si**

Evalúe  $x_{new}$  en función de  $f(x_{new})$ .

Si (aleatorio  $< A_i$  &&  $f(x_{new}) < f(x'_i) \rightarrow find\ min$  )  
 $f(x_{new}) > f(x'_i) \rightarrow find\ max$  )

Actualice soluciones  $x_{new}$  to  $x_i^{t+1}$

**Finalizar si**  $\left[ \begin{array}{l} f(x_{new}) < f(x_s) \rightarrow find\ min \\ f(x_{new}) > f(x_s) \rightarrow find\ max \end{array} \right]$   
 $x_{new}$  se transforma en  $x_s$

Calcule error E.

**Finalizar mientras**

Algunos de los cambios realizados al algoritmo, están relacionados con el recorrido aleatorio definido por la ecuación (5), que se transforma en ecuación. (8). Se observó que el proceso de búsqueda es optimizado, con el uso de esta nueva ecuación.

**2.2 Resultados Experimentales**

Para el análisis de AMV aplicado a la ID, se ejecutaron varias pruebas, sin embargo solo algunas fueron incluidas en este artículo, debido a restricciones de espacio. Se emplearon un par de ejemplos realizados (Maldonado & Chio 2004), con el fin de observar la capacidad del algoritmo en encontrar una respuesta adecuada. Por consiguiente, se realizó un análisis real de un sismo, empleando información de un acelerograma instalado en Chile, el cual registró un sismo en el año 2010 (Facultad de Ingeniería Civil - Universidad de Chile n.d.). Para cada situación, se informó el tiempo recorrido, empleando un computador con las siguientes especificaciones: AMD Phenom II X6 @ 2.8 GHz, 8 GB RAM, que trabaja con Windows 7 Profesional. Para cada experimento, se modificó el número de murciélagos virtuales, con el fin de apreciar su efecto sobre la respuesta.

**2.2.1 Situación N°1. Función Armónica**

La excitación producida dentro del sistema, con un único grado de libertad, es representada por la Expresión (9).

$$P(t) = 1500sen(12.566t) \tag{9}$$

Los datos entregados al software son los siguientes/The data given to the software is as follows:

Mass= 7.6 Ton  
 K= 30000 KN/m  
 $\zeta=5\%$

Como grupo de control, se empleó la respuesta analítica encontrada en (Maldonado & Chio 2004). En este caso, el mejor desplazamiento es  $u(t)=5.2\ cm$ , que se manifestó en un tiempo de 0,085 segundos. La última columna de la Tabla 1 muestra el tiempo requerido para finalizar el proceso. El trazado de la respuesta dinámica es generado empleando 200 puntos. La Tabla 1 resume los resultados de las simulaciones con diferentes cantidades de murciélagos.

**end if**

Evaluate  $x_{new}$  in the function,  $f(x_{new})$ .

if (rand  $< A_i$  &&  $f(x_{new}) < f(x'_i) \rightarrow find\ min$  )  
 $f(x_{new}) > f(x'_i) \rightarrow find\ max$  )

Update solutions  $x_{new}$  to  $x_i^{t+1}$

**if**  $\left[ \begin{array}{l} f(x_{new}) < f(x_s) \rightarrow find\ min \\ f(x_{new}) > f(x_s) \rightarrow find\ max \end{array} \right]$   
 $x_{new}$  becomes  $x_s$

Calculate error E.

**end while**

Some of the changes performed on the algorithm are related to the random walk defined by equation (5), which becomes equation (8). It was observed that by using this new equation, the search process is enhanced.

**2.2 Experimental results**

For analyzing VBA applied to DI, several tests were run, but only some of them are reported due to space limitations. A couple examples shown in (Maldonado & Chio 2004) are used, in order to observe the algorithm's ability to find an appropriate answer. Following that, an analysis on a real-life seism is performed, using data of an accelerometer set in Chile, which registered a seism from 2010 (Facultad de Ingeniería Civil - Universidad de Chile n.d.). For each situation, the run time is reported, using a computer with the following specifications: AMD Phenom II X6 @ 2.8 GHz, 8 GB RAM, working under Windows 7 Professional. For each experiment, the number of virtual bats were varied, in order to see its effect on the answer.

**2.2.1 Situation N°1. Harmonic function.**

The excitation produced into the system with a single degree of freedom, is presented through Expression (9).

As a control group, the analytical answer found in (Maldonado & Chio 2004) is used. In this case, the best displacement is  $u(t)=5.2\ cm$  manifested at a time of 0.085 seconds. The last column of Table 1 shows the time required to finish the process. The plot of the dynamic response is generated using 200 points. Table 1 summarizes the results for simulations with different number of bats.

La Figura 1 muestra el comportamiento a través del tiempo, entregado por el algoritmo.

Figure 1 shows the behavior of displacement through time, provided by the algorithm.

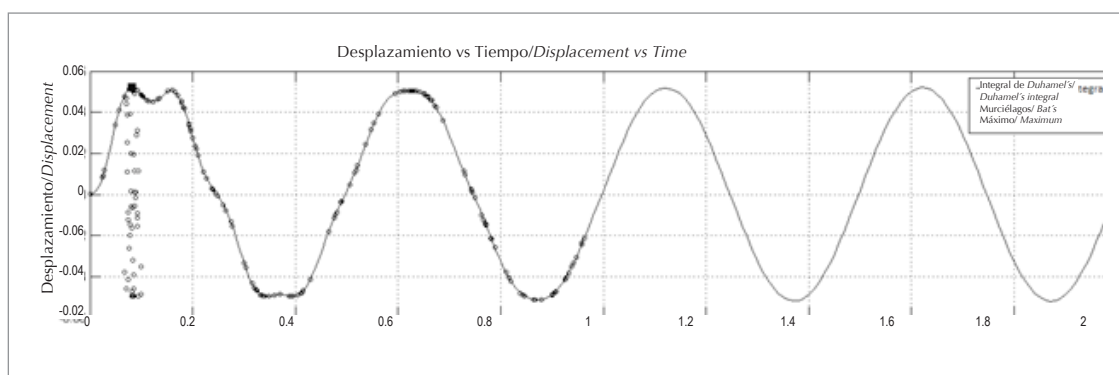
**Tabla 1** Resumen de la cantidad de simulaciones para la situación N°1

**Table 1.** Summary of the number of simulations for the first situation

$p(t) = 1500\text{sen}(12.566t)$									
(Mejor desplazamiento 5,2 cm para $t=0.085$ /The best displacement 5,2 cm for $t= 0,085$ )									
Para 200 puntos en el gráfico/For 200 points on the graph									
Murciélagos/Bat's		Desplazamiento [cm]/Displacement [cm]					Error [%]	Tiempo/time [s]	[s]
		1	2	3	4	5 Promedio/Average			
10	U(t)	5.070	4.894	4.969	5.151	4.989	5.015	3.70	4.714
	t	0.160	0.170	0.148	0.091	0.095	0.133		
	T	4.656	4.656	4.902	4.655	4.701	4.714		
50	U(t)	5.061	5.213	5.213	5.173	5.075	5.147	1.03	4.750
	t	0.162	0.079	0.068	0.077	0.161	0.109		
	T	4.765	4.758	4.753	4.724	4.752	4.750		
100	U(t)	5.160	5.214	5.196	5.196	5.180	5.189	0.21	4.816
	t	0.065	0.081	0.076	0.084	0.085	0.078		
	T	4.868	4.797	4.864	4.778	4.771	4.816		
150	U(t)	5.215	5.212	5.132	5.206	5.205	5.194	0.12	4.900
	t	0.076	0.068	0.894	0.088	0.084	0.242		
	T	4.902	4.896	4.887	4.902	4.911	4.900		
200	U(t)	5.198	5.209	5.214	5.206	5.215	5.208	0.16	4.980
	t	0.089	0.088	0.081	0.093	0.082	0.086		
	T	4.886	4.982	4.979	4.976	4.975	4.980		

Máximo Local/ Local Maximum

Máximo global/ Global Maximum



**Figura 1.** Situación N°1 con 100 murciélagos. Desplazamiento en [m] y tiempo en [s]

**Figure 1.** Situation 1 with 100 bats. Displacement in [m] and time in [s]



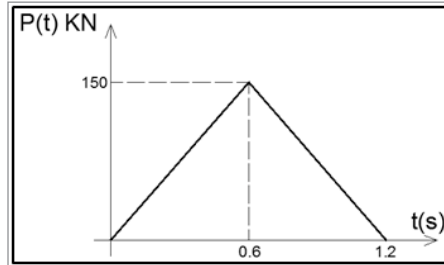
**2.2.2 Situación N° 2. Función Triangular**

En este caso se empleó una función definida por secciones, para excitar el sistema. Su definición se puede apreciar en la Ecuación (10) y está dibujada en la Figura 2.

**2.2.2 Situation N° 2. Triangle function**

In this case, a function defined by sections is used to excite the system. Its definition can be seen in equations (10), and it is plotted in Figure 2.

$$P(t) = \left\{ \begin{array}{l} P(t)_1 = \frac{150}{0.6} t \rightarrow 0 \leq t \leq 0.6 \text{ s} \\ P(t)_2 = -\frac{150}{0.6} t + 300 \rightarrow 0.6 < t \leq 1.2 \text{ seg.} \end{array} \right\} \quad (10)$$



**Figura 2.** Señal de excitación para la segunda situación  
**Figure 2.** Excitation signal for the second situation

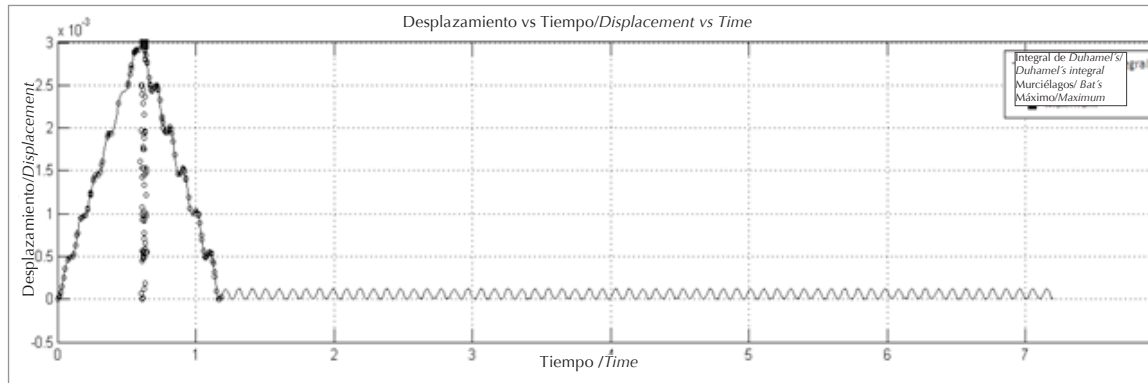
Una vez más, las respuestas proporcionadas por (Maldonado & Chio 2004) fueron empleadas como grupo de control. En este caso, el mejor desplazamiento fue  $u(t)=0.293$  cm para un período de tiempo de 0,6 segundos, aproximadamente. La Tabla 2 resume los resultados alcanzados con el algoritmo, encontrándose en la última columna el tiempo que demora en converger una respuesta. El trazado de la respuesta, con 100 puntos, es mostrado en la Figura 3. Los datos suministrados al algoritmo fueron los siguientes:

Once again, answers from (Maldonado & Chio 2004) were used as a control group. In this case, the best displacement was  $u(t)=0.293$  cm for a time of 0.6 seconds, approximately. Table 2 summarizes the results achieved with the algorithm, finding in the last column the time it took to converge to an answer. A plot of the response, with 100 dots is shown in Figure 3. The data fed to the algorithm was the following:

Mass= 12 Ton  
K= 50000 KN/m  
 $\zeta=0\%$ (Sin amortiguación)/(no damping).

**Tabla 2** Resumen de la cantidad de simulaciones para la situación N°2  
**Table 2.** Summary of the results for Situation N° 2

Ejercicio 2/Exercise N°. 2									
Mejor desplazamiento 0,293 cm para t=0.6/The best displacement 0,293 cm for t= 0,06									
Para 100 puntos en el gráfico/For 100 points un the graph									
Murciélagos/Bat's		Desplazamiento [cm]/Displacement [cm]					Error [%]	Tiempo/time [s]	
		1	2	3	4	5 Promedio/Average			
10	U(t)	0.293	0.256	0.248	0.242	0.298	0.268	9.52	33.079
	t	0.631	0.662	0.508	0.483	0.620	0.581		
	T	32.354	33.051	32.729	33.591	33.670	33.079		
50	U(t)	0.289	0.296	0.298	0.298	0.293	0.295	0.63	33.212
	t	0.636	0.597	0.632	0.614	0.597	0.615		
	T	33.204	33.075	33.392	33.207	33.183	33.212		
100	U(t)	0.299	0.295	0.299	0.297	0.299	0.297	1.50	33.549
	t	0.617	0.629	0.616	0.594	0.608	0.613		
	T	33.815	33.321	33.539	33.573	33.499	33.549		
150	U(t)	0.299	0.298	0.298	0.299	0.298	0.298	1.80	33.497
	t	0.618	0.640	0.617	0.618	0.620	0.623		
	T	33.541	33.695	33.482	33.465	33.304	33.497		
200	U(t)	0.299	0.298	0.298	0.299	0.299	0.298	1.82	33.963
	t	0.623	0.621	0.621	0.618	0.628	0.622		
	T	34.010	33.725	34.358	34.069	33.652	33.963		



**Figura 3.** Situación N°2 con 100 murciélagos virtuales. Desplazamiento en [m] y tiempo en [s]

**Figure 3.** Situation No. 2 with 100 bats. Displacement in [m] and time in [s]

### 2.2.3 Situación N°3. Acelerograma de Terremoto. Sismo en Chile (Terremoto del Maule)

La excitación empleada en esta situación fue tomada de los datos de un terremoto de 8,8 grados, conocido como el terremoto del Maule, ocurrido el 27 de febrero de 2010. Algunas especificaciones de este evento sísmico son: aceleración máxima 465.484 cm/s<sup>2</sup>, velocidad máxima 27.707 cm/s y desplazamiento máximo del terreno 7.001 cm. El algoritmo es alimentado con datos de aceleraciones informadas por (Facultad de Ingeniería Civil - Universidad de Chile n.d.), y con los siguientes valores estructurales:

$$\begin{aligned} \text{Mass} &= 7.6 \text{ Ton} \\ K &= 30000 \text{ KN/m} \\ \zeta &= 5\% \end{aligned}$$

Puesto que no existen datos comparativos para este evento, se empleó el AMV para observar la respuesta y chequear el tiempo de recorrido necesario para calcular el desplazamiento máximo. La Figura 4 muestra los datos sísmicos, así como la respuesta del algoritmo para el desplazamiento del tiempo, en función del tiempo. Nuevamente la Tabla 3 muestra los resultados para las diferentes cantidades de murciélagos.

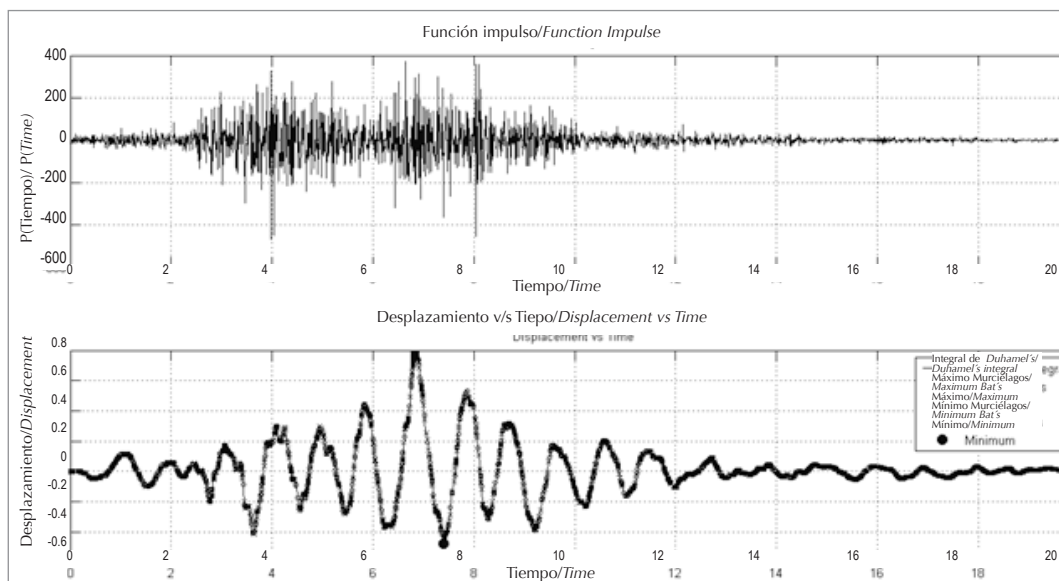
### 2.2.3 Situation N°3. Earthquake accelerogram. Seism in Chile (Maule's earthquake)

The excitation used in this situation is taken from data of an 8.8 quake, known as Maule's earthquake, which happened on February 27th, 2010. Some of the specifications of this seismic event are: peak acceleration of 465.484 cm/s<sup>2</sup>, peak velocity of 27.707 cm/s and a peak ground displacement of 7.001 cm. The algorithm is fed with a file containing the accelerations reported by (Facultad de Ingeniería Civil - Universidad de Chile n.d.), and the following structural values:

Since this situation does not have comparative data, BA was used to observe the response and to check the run time required for calculating the maximum displacement. Figure 4 shows the seismic data, as well as the algorithm's response for time displacement as a function of time. Once again, Table 3 summarizes the results for different number of bats.

**Tabla 3.** Resumen de los resultados para la situación N° 3  
**Table 3.** Summary of the results for situation N° 3

Resultados con el terremoto de Chile. Terremoto del Maule/Result with Chile earthquake, Maule Earthquake							
Para 100 puntos en el gráfico/For 100 points in the graph							
Murciélagos/Bat's	Desplazamiento [cm]/Displacement [cm]					Tiempo/time [s]	
		1	2	3	4		Promedio/Average
100	U(t) Max	0.709	0.762	0.716	0.736	0.443	42.734
	t Max	6.802	6.867	6.910	6.898	7.979	
	U(t) Min	-0.395	-0.397	-0.415	-0.408	-0.388	
	t Min	3.658	7.466	7.436	3.648	7.353	
	T	42.741	42.647	42.764	42.828	42.690	
200	U(t) Max	0.580	0.762	0.762	0.602	0.755	43.320
	t Max	6.755	6.819	6.866	6.770	6.832	
	U(t) Min	-0.477	-0.411	-0.470	-0.478	-0.423	
	t Min	7.405	3.641	7.392	7.404	7.430	
	T	43.462	43.443	43.140	43.162	43.391	
500	U(t) Max	0.735	0.755	0.761	0.765	0.718	45.681
	t Max	6.907	6.854	6.873	6.825	6.800	
	U(t) Min	-0.437	-0.398	-0.437	-0.479	-0.459	
	t Min	7.423	7.466	7.375	7.403	7.387	
	T	46.002	45.631	45.721	45.355	45.696	
1000	U(t) Max	0.765	0.759	0.766	0.762	0.764	48.577
	t Max	6.842	6.865	6.819	6.834	6.829	
	U(t) Min	-0.469	-0.479	-0.473	-0.442	-0.479	
	t Min	7.409	7.397	7.393	7.379	7.397	
	T	48.763	48.598	48.468	48.371	48.683	
1500	U(t) Max	0.766	0.763	0.761	0.762	0.764	52.370
	t Max	6.834	6.831	6.872	6.864	6.829	
	U(t) Min	-0.466	-0.443	-0.480	-0.477	-0.477	
	t Min	7.390	7.380	7.402	7.405	7.396	
	T	52.170	52.326	52.727	52.264	52.364	
2000	U(t) Max	0.766	0.766	0.762	0.762	0.766	55.601
	t Max	6.825	6.823	6.865	6.831	6.833	
	U(t) Min	-0.460	-0.479	-0.465	-0.478	-0.459	
	t Min	7.388	7.397	7.390	7.396	7.414	
	T						
	Máximo Local/Local Maximum						
	Máximo global/Global Maximum						
	Local Minimum						



**Figura 4.** Situación N°3 con 2000 murciélagos virtuales. Desplazamiento en [m] y tiempo en [s]  
**Figure 4.** Situation N° 3 with 2000 virtual bats. Displacement in [m] and time in [s]

### 3. Conclusiones

Durante esta investigación fue posible observar que los tres modelos, que emplean la Integral de Duhamel, alcanzaron resultados altamente satisfactorios para el cálculo del desplazamiento máximo de una estructura, con un único grado de libertad. El algoritmo de murciélago virtual es bastante preciso en encontrar el máximo y mínimo de la función, por lo que puede ser empleado cuando un problema en particular así lo requiera. Los tiempos recorridos del algoritmo fueron relativamente breves para el número de murciélagos empleado durante las pruebas.

Se observa que el algoritmo tiende a permanecer en los extremos locales, cuando el dominio de búsqueda es grande, en comparación al número de murciélagos empleado (es decir, el algoritmo murciélago trabaja mejor cuando busca por partidas, al menos para esta aplicación). De cualquier manera y gracias a su naturaleza estocástica meta-heurística, el algoritmo encuentra el valor óptimo con mucha rapidez en el caso de la integral de Duhamel. Los resultados resumidos por las tablas muestran que la respuesta entregada por el algoritmo no difiere demasiado y que se estabiliza cerca de una respuesta, en un tiempo breve. Debido a lo anterior, se recomienda continuar investigando este algoritmo, y emplearlo para resolver problemas de relativa complejidad matemática.

### 3. Conclusions

*During this research, it was possible to observe that for all of the three exemplary cases with Duhamel's Integral, highly satisfactory results were achieved when calculating the maximum displacement of a structure with a single degree of freedom. The virtual bat algorithm is quite precise when finding the maximum and minimum of a function, so it can be used as a particular problem requires it. The run times of the algorithm were relatively short for the number of bats used during the tests.*

*It was observed that the algorithm tends to remain in local extremes when the search domain is big in comparison to the number of bats used (i.e. the bat algorithm works better, at least for this application, when searching in packs). In any case, and thanks to it being a stochastic metaheuristic, the algorithm finds the optimum, in this case Duhamel's Integral, very quickly. The results summarized in the tables show that the answer reported by the algorithm does not differ too much, and that it stabilizes around an answer in a short time. Because of this, it is recommended to keep working on it, using it for solving problems with a relatively high mathematical complexity.*

### 4. Referencias/References

- Facultad de Ingeniería Civil - Universidad de Chile (2012)**, Base de Datos Sismo de Chile. Available at: <http://terremotos.ing.uchile.cl/> [Accessed November 1].
- Gandomi A.H. et al. (2012)**, Bat algorithm for constrained optimization tasks. *Neural Computing and Applications*. Available at: <http://www.springerlink.com/index/10.1007/s00521-012-1028-9> [Accessed November 24, 2012].
- Kennedy J. and Eberhart R.C. (1997)**, A Discrete Binary Version of the Particle Swarm Algorithm. In 1997 IEEE International Conference on Systems, Man, and Cybernetics. *Computational Cybernetics and Simulation*. IEEE, pp. 4104–4108. Available at: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=637339> [Accessed December 11, 2012].
- Maldonado E. and Chio C.G. (2004)**, Análisis Sísmico de Edificaciones 1st ed., Ediciones Universidad Industrial de Santander.
- Matlab Central, Bat Algorithm (demo) (2012)**, Available at: <http://www.mathworks.com/matlabcentral/fileexchange/37582-bat-algorithm-demo> [Accessed November 1].
- Nakamura R.Y.M. et al. (2012)**, BBA: A Binary Bat Algorithm for Feature Selection. In XXV SIBGRAPI - Conference on Graphics, Patterns and Images. Ouro Preto, pp. 291–297. Available at: <http://www.decom.ufop.br/sibgrapi2012/eeproceedings>.
- Parsopoulos K.E. and Vrahatis M.N. (2010)**, Particle Swarm Optimization and Intelligence: Advances and Applications,
- Paz M. (1992)**, Dinámica Estructural 3rd ed., Editorial Reverté, S.A.
- Ponnappa M. and Pupong P. (2012)**, Solving Multi-Stage Multi-Machine Multi-Product Scheduling Problem Using Bat Algorithm. In 2nd International Conference on Management and Artificial Intelligence IPEDR. Singapore: IACSIT Press, pp. 98–102.
- Ramos C.C.O. et al. (2011)**, A Novel Algorithm for Feature Selection Using Harmony Search and its Application for Non-Technical Losses Detection. *Computers & Electrical Engineering*, 37(6), pp.886–894. Available at: <http://linkinghub.elsevier.com/retrieve/pii/S0045790611001479> [Accessed December 11, 2012].
- Yang X. (2010)**, A New Metaheuristic Bat-Inspired Algorithm. In International Workshop on Nature Inspired cooperative Strategies for Optimization (NCSO 2010). Granada: Springer, pp. 65–74.
- Yang X. (2011)**, Bat Algorithm for Multiobjective Optimization. *International Journal of Bio-Inspired Computation*, 3(5), pp.267–274.
- Yang X.-S. and Gandomi, A.H. (2012)**, Bat Algorithm: A Novel Approach for Global Engineering Optimization. *Engineering Computations*, 29(5), pp.464–483.
- Yang Xing-She. (2010)**, Nature-Inspired Metaheuristic Algorithms 2nd ed., United Kingdom: Luniver Press.