International Journal of Computational Science and Mathematics.
ISSN 0974-3189 Volume 3, Number 2 (2011), pp. 173-181

http://www.irphouse.com

Determining Earthquake Locations in NW Himalayan Region: An Application of Particle Swarm Optimization

Kusum Deep[1], Anupam Yadav[1] and Sushil Kumar[2]

[1]*Department of Mathematics, Indian Institute of Technology,*
Roorkee, Uttarakhand, India
[2]*Wadia Institute of Himalayan Geology, Dehradun, Uttarakhand, India*
E-mail: anupuam@gmail.com, kusumdeep@iitr.ernet.in,
sushil_rohella@yahoo.co.in

Abstract

Inversion problems in seismology deal with the estimation of the location of an earthquake from the observations of the arrival times of the body waves. This problem is modeled as a non-linear optimization problem in which the objective function to be minimized is the discrepancy between the observed and the calculated travel times. This paper attempts to determine the seismic location in the upper mantle of the Earth's crust using a new nature inspired optimization technique named "particle swarm optimization". With the help of this technique, the location of the earthquakes in the northern Himalayan and Hindu Kush region is determined. The location of the Earthquakes up to the depth 100 Km are considered. An advance version of PSO namely LXPSO is used for the inversion of data.

Keywords: Particle Swarm Optimization (PSO), LXPSO, Earthquake, Hypocenter.

Introduction

Locating Earthquakes is one of the oldest problems in seismology and leftovers an area of active research. The problem is complicated by the nonlinear dependence of seismic travel times on location, incomplete knowledge of three dimensional velocity structures along the source receiver path and difficulties associated with the inadequate station coverage and outliers in the observed travel time picks. Now a day, with increased computer capabilities a lot of new methods and algorithms are being developed for Earthquake locations.

Literature

There are several methods offered for finding the Earthquake location depending upon the different velocity models of seismic waves in the Earth's crust. The majority of them in the standard catalogs are still using conventional least square methods, one reason for this is a desirable conservatism in the production of catalogs, whose value is derived from their consistency, which makes it possible to compare seismicity patterns at different times without the possible biasing effects of a change in the location methods.

The most challenging fact while finding an Earthquake location is the heterogeneity of the Earth's crust. Due to this heterogeneity of Earth crust it is a very complicated for seismologists to study the average crustal velocity of the seismic waves. There are more than a few velocity models developed for the average crustal velocity of the seismic waves at different depths of the Earth crust in the Himalayan region [e.g., Roecker SW, 1982[10]; Kalia et al. 1969[4]; Matveyeva and Lukk, 1968[7]; Ram and Mereu, 1977[9]; Kumar and Sato, 2003[11]]. According to Kumar and Sato, 2003[11], the velocity of the compressional waves with in the depth 0- 15 Km is 5.2 Km/sec and 15-40 Km is 5.89 Km/sec. According to Kalia et al. 1969 the velocity of compressional wave within the depth 40-70 Km is 8.14 Km/sec, 70-85 Km is 8.32 Km/sec and 85-100 is 8.29 Km/sec.

One way to solve the problem of determining earthquake location is to model the problem as a nonlinear optimization problem in which the objective function to be minimized is the discrepancy between the observed travel times and calculated travel times. Earlier Shanker et al. 1991[6] used a random search technique to solve the problem.

In this paper a new heuristic technique namely Particle Swarm Optimization is used to solve the problem of determination of Earthquake location. In fact two versions of PSO [3] are used – the first one is the Standard PSO of Kennedy and Clerc (2006) [8] and the second is a new LXPSO of Bansal et al (2009) [2]. The method is validated on real life data for NW Himalayas.

The paper is organized as follows. In section 3 we have defined mathematical models then a brief discussion on particle swarm optimization algorithms and their results on seismic data.

Mathematical Model

Conventionally the mathematical model may be described in two parts, the first one is forward problem model and second one is inversion problem model. We will discuss each one by one.

Forward Problem Model

This model can also be called as travel time model which gives the travel time of the compressional wave in the different layers of the Earth crust. As stated above the travel time of the compressional waves in the different layers of the Earth using the velocity model (1) will be calculated for different depths, then the travel time of seismic waves in each layer to get the total travel time for the waves will be added to reach at the observational stations on the surface of the Earth from focus.

Let the parameters of the hypocentre be (x_i, y_i, z_i), where they represent the coordinate values of latitude, longitude and depth of the preliminary hypocentre. (x_j, y_j) be the latitude and longitude of the stations. v_l is the average crustal velocity of the Compressional waves in the l^{th}layer of the Earth. The theoretical travel time and epicentral distances according to Yang et al. 2007[12] are given by

$$t_{ij} = \frac{\sqrt{\Delta_{ij}^2 + Z_i^2}}{v_l} \quad (1)$$

Where

$$\Delta_{ij} = 111.199\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 cos^2 \frac{(x_i - x_j)}{2}}$$

Inverse problem model

In this model the values of hypocentral parameters will be calculated inversely by minimizing a root mean square function of calculated and observed travel times. Let C_k and O_k represents the calculated and observed travel times. Equation (2) gives the root mean square function of C_k and O_k which we have to minimize.

$$Minimize\ f = \{\sum_{k=1}^{n} (C_k - O_k)^2\}^{1/2} \quad (2)$$

For a single Earthquake our C_k will be

$$C_k = \frac{\sqrt{\Delta_k^2 + Z_k^2}}{v_l} \quad (3)$$

Where

$$\Delta_k = 111.199\sqrt{(x - x_i)^2 + (y - y_i)^2 cos^2 \frac{(x - x_i)}{2}}$$

Here (x, y, z) is the hypocentre of the Earthquake. In order to validate the model, real data from NW Himalayas is considered. Therefore to physically validate the model it is necessary to incorporate the following restrictions:

Latitude lower < x < Latitude upper;

Longitude lower < y < Longitude upper;

Depth lower < z < Depth upper.

As we are finding the hypocenters in the NW Himalayan and Hindukush region. These restriction limits are taken as

$22^0 < x < 36^0$; $68^0 < y < 98^0$; $0 < z < 100$(Kms)

The above nonlinear optimization problem will be solved by the standard particle swarm optimization of Kennedy and Clerc, 2006[8] and a newly developed version of particle swarm optimization namely LXPSO of Bansal et al. 2009[2]. These are explained in the next section.

Particle Swarm Optimization

Particle swarm optimization is a nature inspired evolutionary search technique which is probabilistic in nature. It is inspired by the social behaviour of animals such as bird's flocking and fish schooling. It was jointly proposed by Kennedy and Eberhart, 1995[3]. It simply uses the learning, information sharing and position updating strategy and very simple to implement. For better understanding mathematically it can be defined as follows:

For a D-dimensional search space, the nth particle of the swarm at time step t is represented by D-dimensional vector $x_n^t = (x_{n1}^t, x_{n2}^t, ..., x_{nD}^t)^T$, the velocity of the particle at time step *t* is denoted by $V_n^t = (V_{n1}^t, V_{n2}^t, ..., V_{nD}^t)^T$ let the best visited position of the particle at time step *t* is $P_n^t = (P_{n1}^t, P_{n2}^t, ..., P_{nD}^t)^T$. Let '*g*' is the index of the best particle in the swarm. The position of the particle and its velocity is being updated using the following equations:

$$V_{nd}^{t+1} = V_{nd}^t + c_1 r_1 (P_{nd}^t - x_{nd}^t) + c_2 r_2 (p_{gd}^t - x_{nd}^t) \quad (4)$$

$$x_{nd}^{t+1} = x_{nd}^t + V_{nd}^t \quad (5)$$

here $d = 1, 2, 3, ...D$ represents the dimension and $n = 1, 2, 3, ...S$ represents the particle index. c_1 and c_2 are the constants and r_1 and r_2 are random variables with unifrom distribution between 0 and 1. Equation (4) and (5) define the classical version of PSO. The basic PSO algorithm by Kenedy and Eberhart, 1995[3] is given by :

Create and initialize a D - dimensional swarm, S
For t =1 to the maximum_iterations
For n =1 to S,
For d = 1 to D,
Apply the velocity update equation (4)
Update position using equation (5)
End for d ;
Compute fitness of updated position ;
If needed update the historical information about P_n *and* P_g .
End for n ;
Terminates if P_g *meets problem reauirements;*
End for t ;

The LXPSO Algorithm

Every day many improved versions of PSO's are appearing in the literature. One such improved PSO is the LXPSO of Bansal et al. 2009[2]. Herein, PSO is hybridized by incorporating the Laplace's Crossover (Earlier designed for Genetic Algorithm by Deep and Thakur, 2007). The supremacy of LXPSO[2] over SPSO [8] is well established in Bansal et al. 2009, on a set of Standard benchmark problems. In LXPSO[2] algorithm we will use a term Lapalcian particle, the description of same is as follows:

Laplace's crossover proposed by Deep and Thakur, 2007[5] follows Laplace's Distribution. This parent centric operator is called Laplace's Operator (LX). Here two offspring $y_1 = (y_{11}, y_{12,} ... y_{2D})$ and $y_2 = (y_{21}, y_{22}, ... y_{2D})$ are generated from a pair of parents $x_1 = (x_{11}, x_{12,} ... x_{2D})$ and $x_2 = (x_{21}, x_{22}, ... x_{2D})$ using LX as follows:

First a uniform distributed random number $u_d \in (0,1)$ is generated. Then from Laplace's distribution function; the ordinate β_d is calculated so that the area under probability curve excluding area a (location parameter) to β_d is equal to chosen random number u_d.

$$\beta_d = \begin{cases} a - b\log_e(1-2u_d), u_d \le \frac{1}{2} \\ a - b\log_e(2u_d - 1), u_d > \frac{1}{2} \end{cases}$$

Here b is called the scale parameter. The offspring are then given by the equations:

$$y_{1d} = x_{1d} + \beta_d \left| x_{1d} - x_{2d} \right|$$

$$y_{2d} = x_{2d} + \beta_d \left| x_{1d} - x_{2d} \right|$$

$$d = 1, 2, 3, ... D$$

based on Laplacian operator as described above two particles are formed. The best particle (in terms of fitness) is selected. This new particle is called Lapalcian Particle.

LXPSO algorithm is as follows:

Create and initialize a D-dimensional swarm, S
For t = 1 to the maximum_iterations,
For n = 1 to S,

For d = 1 to D,

Apply the velocity update equation (4) *and update the position using equation* (5)
End for d;
Compute fitness of updated position;
If needed, update historical information for P_n *and* P_g *;*

End-for-n;

Select two random particles from the current swarm for interaction. Generate the Lapalcian particle as a result of this interaction.Replace the worst particle in the swarm with Lapalcian particle;

Compute fitness of Lapalcian Particle;

If needed, update historical information for P_n *and* P_g *;*

Terminate if P_g *meets problem requirements;*

End for t;

Selection of Parameter values for SPSO and LXPSO

Dimension (D) of the search space is 3, i.e. *D*=3. The decision variables in algorithm are latitude, longitude and the depth of the hypocentre. The selection of parameter is done according to Kennedy and Clerc, 2006[8]. The swarm size is 13 [10+(int)(2*sqrt(*D*))] , $c_1 = c_2 = 0.5 + \log(2)$. Number of iterations are 200.

The results of Standard PSO are compared with LXPSO[2]. Number of iterations and the values of used parameters values in both the versions of PSO are same. This ensures the meaningful comparison of both the algorithms.

Testing of SPSO [8] and LXPSO on the Earthquake data of Himalayan and Hindu Kush Region

This test is based on the real observed data of Earthquake on 25 September 2008 in Hindu Kush and NW Himalayan region, an Earthquake is recorded of intensity 5.0 on Richter scale whose observed location is 28.889^0, 85.013^0 and depth of 111.6 Km [Table 1]. Based on this data we have calculated the travel time in each layer of the Earth at different depths to the ten different stations by using equation (3) from which this data is observed. Table 2 gives the detailed calculation of travel times different layers. Using the travel time data we minimize the function f given by eq. (2).

On observing the results presented in Table1. and Table 2. it is observed that whien the forward problem was solved using the hypocentral parameters in the NW Himalayan region and the calculated travel time were used to solve the inverse problem the the results obtained matched very well the actual earthquake parameters. This is so with both the versions of PSO used. However on observing closely the Table 3. it is found that the objective function value obtained by LXPSO[2] is better than that obtained by SPSO[8]. This determines the supremacy of LXPSO[2] over SPSO[8] for solving this problem.

Fig 1. shows the way in which the value of the objective function decreases as and how the generation increase. The comparison of the convergence graph of SPSO[8] and LXPSO[2] are superimposed. This again shows the LXPSO[2] converges much faster than SPSO[8].

Table 1: Location of earthquake and the locations of observations stations.

Hypocenter	Stations	
28.889^0 85.013^0 111.6 km	30.00^0	70.00^0
	30.14^0	79.20^0
	30.97^0	77.86^0
	30.71^0	77.25^0
	31.10^0	79.61^0
	30.54^0	78.10^0
	29.71^0	78.43^0
	30.49^0	77.58^0
	30.53^0	77.73^0
	30.33^0	78.74^0

Table 2: Travel time of Compressional waves at different depths of the Earth's crust.

LXPSO				Standard PSO			
$\boldsymbol{x}^0$	$\boldsymbol{y}^0$	$\boldsymbol{z}$(Km)	$\boldsymbol{f}$	$\boldsymbol{x}^0$	$\boldsymbol{y}^0$	$\boldsymbol{z}$(Km)	$\boldsymbol{f}$
28.0087	84.0105	110.25	0.000279	28.0056	84.0005	110.16	0.000301
28.0079	84.0109	110.10	0.000391	28.0074	84.0029	110.01	0.000422
28.0010	84.0583	109.65	0.000853	28.0001	84.0097	109.87	0.000934
28.0021	84.0676	109.86	0.000712	28.0019	84.0174	109.12	0.000937
28.0053	84.0873	110.78	0.000842	28.0007	84.0652	110.23	0.000600
28.0015	84.0423	110.64	0.000568	28.0001	84.0134	110.33	0.000832
27.8800	84.0202	110.65	0.000704	27.8898	84.0203	110.09	0.000826
28.4300	84.0161	109.12	0.000627	28.4334	84.0304	109.15	0.000921
28.1276	85.0283	110.14	0.000740	28.1245	85.0403	110.13	0.000844
28.3256	85.0037	110.34	0.000376	28.3212	85.0006	110.24	0.000632

Table 3: Results using SPSO and LXPSO shows the minimize value of $\boldsymbol{f}$ and resultant hypocentral parameters.

Travel Times (Sec) in different depths (in Kms)					
0-15	15-40	40-75	75-85	85-100	Total
3.70972	4.71768	3.97535	2.31858	2.32697	17.04830
3.81867	4.78496	4.01723	2.38667	2.39531	17.40284
3.79631	4.77107	4.00857	2.37269	2.38128	17.32992
3.79271	4.76884	4.00718	2.37044	2.37902	17.31819
3.78329	4.76300	4.00354	2.36456	2.37311	17.28750
3.78739	4.76554	4.00512	2.36712	2.37568	17.30085
3.80601	4.77709	4.01232	2.37876	2.38736	17.36154

3.81897	4.78515	4.01734	2.38686	2.39550	17.40382
3.80686	4.77762	4.01265	2.37929	2.38790	17.36432
3.80147	4.77427	4.01056	2.37592	2.38452	17.34674

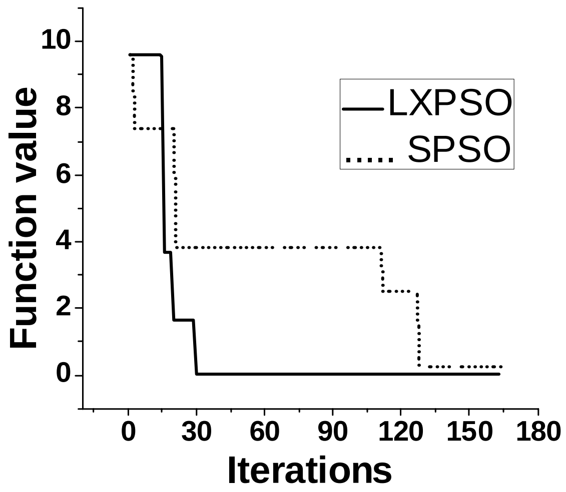

Figure 1: Comparison of convergence of LXPSO and SPSO[8].

Conclusion

The goal of this paper is to determine the hypocentral parameters. Inverse problem is modeled as a non-linear optimization problem in which the objectives function to be minimized is the root mean square of the observed travel time and calculated travel time of p waves. The model is validated by taking a real earthquake data of NW Himalayan region. The problem is solved by two version of particle swarm optimization technique, standard PSO and LXPSO. Form the numerical results and graphical results it is observe that both the versions of PSO are able to give good results. Further LXPSO provides a better accuracy of results in comparison to SPSO [8].

It is recorded that this approach of PSO is a novel way to solve the inversion problem of hypocentral parameters.

References

[1] Hang Dong-xue and Wang Gai-yun. Application of particle swarm optimization to seismic location. *Third Int Conf of Genetic and Evolutionary Computing*, 0: pp 641-644, 2009.

[2] J.C.Bansal, Veeramachaneni K., Deep Kusum, and L. Osadciw. Information sharing strategy among particles in particle swarm optimization using laplacian operator. *Swarm Intelligence Symposium, SIS ’09. IEEE*, 30: 30–36, 2009.

[3] James Kennedy and Eberhart R. Particle swarm optimization. *IEEE International Conference on Neural Networks, 1995. Proceedings*, 4: pp 1942 – 1948 , 1995.

[4] K.L. Kaila, Krishna, V. G. Narain, Hari. Upper Mantle velocity structure in the Hindukush region from travel time studies of deep earthquakes using a new analytical method. *Bulletin of the seismological society of America*, 59(5): pp 1949–1967, 1969.

[5] Kusum Deep and Manoj Thakur. A new crossover operator for real coded genetic algorithms. *Applied Mathematics and Computation*, 188(1): pp 895 – 911, 2007.

[6] Kusum Shanker(now Deep) C. Mohan and K.N. Khattri. Inversion of seismological data using a controlled random search global optimization technique. *Tectonophysics*, 198(1):73 – 80, 1991.

[7] Matveyeva MM, Lukk AA. Estimates of the accuracy inconstructing travel time curve for the Pamir-Hindu Kush zone and inthe computer determination of the velocity profile in the upper mantle. *Izv. Acad. Sci. U.S.S.R. Phys. Solid Earth, Engg. Transl*. 8: pp 466-473, 1968.

[8] Maurice clerc and Kennedy J. Standard Particle Swarm Optimization. *http://www.particleswarm.info/Standard_PSO_2006.c*

[9] Ram, A. Mereu, R. F. Lateral variations in upper-mantle structure around india as obtained from Guaribidanur seismic array data. *Geophysical Journal of the Royal Astronomical Society*, 49: pp 87–113, 1977.

[10] S. W. Roecker. Velocity structure of the pamir-hindu kush region: Possible evidence of subducted crust. *J. Geophys. Res*., 87: pp 0148–0227, 1982.

[11] Sushil Kumar and Tamao Sato. Compressional and shear wave velocities in the crust beneath the garhwal himalaya, north india. *Journal of Himalayan Geology*, 24(2): pp 77–85, 2003.

[12] Yang Wen-dong and Xing J. A new seismic location method. *Earthquake and Engineering Vibrations*, 27: pp 20–25, 2007.