ESTIMATING ATMOSPHERIC AREA SOURCE STRENGTH THROUGH PARTICLE SWARM OPTIMIZATION

Eduardo Fávero Pacheco da Luz
Centro Nacional de Monitoramento e Alertas de Desastres N...

Haroldo Fraga de Campos Velho
National Institute for Space Research, Brazil

José Carlos Becceneri
National Institute for Space Research, Brazil

Inverse Problems, Design and Optimization Symposium
Miami, Florida,U.S.A., April 16-18, 2007

ESTIMATING ATMOSPHERIC AREA SOURCE STRENGTH THROUGH PARTICLE SWARM OPTIMIZATION

Eduardo Fávero Pacheco da Luz
Gratuation Program in Applied Computing - CAP
Associate Laboratory for Computing and Applied Mathematics - LAC
National Institute for Space Research - INPE
Sao Jose dos Campos, SP, Brazil
eduardo.luz@lac.inpe.br

Haroldo Fraga de Campos Velho
Associate Laboratory for Computing and Applied Mathematics - LAC
National Institute for Space Research - INPE
Sao Jose dos Campos, SP, Brazil
haroldo@lac.inpe.br

José Carlos Becceneri
Associate Laboratory for Computing and Applied Mathematics - LAC
National Institute for Space Research - INPE
Sao Jose dos Campos, SP, Brazil
becce@lac.inpe.br

Débora Regina Roberti
Department of Physics
Federal University of Santa Maria - UFSM
Santa Maria, RS, Brazil
d_r_roberti@yahoo.com.br

ABSTRACT

Atmospheric area pollutant sources are estimated using Particle Swarm Optimization (PSO). The forward problem is addressed by a Lagrangian particle model to simulate the pollutant dispersion. The inverse results are compared with a previous work that used a deterministic method (quasi-Newton approach). The PSO approach produced a good identification and no regularization operator was employed, showing the robustness and efficacy of the bio-inspired algorithm.

PARTICLE SWARM OPTIMIZATION

The particle swarm optimization is a relatively recent heuristic search method the uses the behavior of biological societies, like birds or fishes to drive the artificial particles in a search pattern that equalizes a global search with a local search, in a way that an optimum, or near to optimum, result is found [1].

There is a socio-cognitive theory that supports the particle swarm model, and it is quite simple. The cultural adaptive process encloses a high-level component, as seen in pattern formation trough the individual and the ability of problem solving, and a low-level component, the individual behavior, that can be summarized in three principles [2]:

- Evaluate;
- Compare;
- Imitate.

The tendency to evaluate stimuli, to classify them as positive or negative is, maybe, the behavioral characteristic more present in several living beings. The learning cannot occur unless the living being can evaluate, distinguish from what is good or not.

The social comparison theory describes the behavior of people comparing each other as a pattern to find a way to improve each one quality.

The imitation behavior gives the living beings the ability to learn through the observations of others attitudes. This behavior is not so much common in the nature as believed in common sense. The imitation behavior is more central to the human society, but it can be used in a way to improve the performance of bio-inspired algorithms.

These three principles can be combined, even in simplified computational entities, providing them the ability to adapt their selves in complex and challenging environments, solving several classes of problems.

PSO solves this problems searching for the optimum in an infinite ($\Re^{\infty}$) or a n-dimensional space of real numbers, symbolized as $\Re^{N}$. In a matter of fact, the infinite space is reduced to the computable space that can be reproduced in personal computers, this fact brings aboard the inconvenience of rounding errors.

Kennedy and Eberhart have proposed a simplified way to allow bird, or particles, to fly in this search space, looking for the optimization of a predetermined function.

The position of a particle in the search space represents the solution of the problem, and this position can be represented by an algebraic vector of real numbers, $\vec{x_i}$, or even binary numbers, for a simpler problem.

At each time step, or iteration of the algorithm, the position of the particle is updated, by the addition of a speed vector, $\vec{v_i}$, in each dimension of the particle, in this way, this change can be summarized as,

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

Where $\vec{x_i}(t-1)$ is the previous position of the particle.

The velocity that is applied to the particle is function of two basic behaviors, that can be designed as,

- a cognitive component and,
- a social component.

The cognitive component regulates the weight that is given to the private information that the particle have. In this case, our particle have a peculiar memory, which allows it to remember where, in the search space it was best positioned.

The social component is band regulated, it allow the particle to share information. In a regular society, information is shared for the good of every one. The same principle is reproduced here. Now, each particle can know, from a certain neighborhood, what is the best position ever found by its neighbors.

The social component can be implemented in two different ways. The first one, designed local best, demands the definition of a fixed neighborhood, where, only from these, the information can be shared. A second one, and most used, introduces the concept of global best, the particle has as its neighbors all the other particles in the swarm, this concept allows the information sharing in a high-level, in the sense that the particle will always knows where the search space the swarm is better positioned.

With this background theory, the definition of velocity, that will be applied in the position of each particle, can be given by

$$\vec{v_i} = \vec{v_i}(t-1) + c_1 rand_i\left(\vec{p_i} - \vec{x_i}\right) + c_2 rand_i\left(\vec{p_g} - \vec{x_i}\right) \quad (2)$$

Where c_1 and c_2 are real numbers that weights the cognitive and social components, respectively, $rand_i$ are random numbers [0,1], $\vec{p_i}$ is the best position achieved by the particle and $\vec{p_g}$ is the best position of the swarm, with $\vec{x_i}$ corresponding to the particle actual position.

At the figure bellow, we can see a concept of this movement:

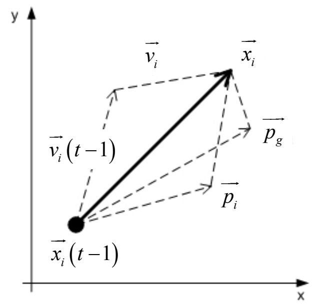

Figure 1. Update scheme for particle position.

A variant of PSO [3] is used in order to bring more stability to the algorithm. A new parameter is used in order to control, in more effective way, the changes between the global search and the local search.

This new parameter, *w*, an inertia parameter, is applied in the previous velocity. And the selection of its value follows an heuristic rule, where:

- If $w < 0.8$: the algorithm works in a local search mode, in a exploration mode;
- If $w > 1.2$: the algorithm works in a exploitation mode, where the global is more explored;
- If $0.8 < w < 1.2$: a balance point, where exploration and exploitation are well divided.

With this new parameter equation (2) can be rewritten to a new one, and in association with equation (1), the heart of PSO is the following set of formulas:

$$\begin{cases} \vec{v_i} = w\vec{v_i}(t-1) + c_1 rand_i\left(\vec{p_i} - \vec{x_i}\right) \\ \quad + c_2 rand_i\left(\vec{p_g} - \vec{x_i}\right) \\ \vec{x_i} = \vec{x_i}(t-1) + \vec{v_i} \end{cases} \quad (3)$$

INVERSE PROBLEM IN ATMOSPHERIC POLUTTION

Direct problem

A Lagrangian scheme can be employed to calculate the concentration of a pollutant

$$C\left(\vec{x},t\right) = \int_{-\infty}^{t}\int S\left(\vec{x_0},t_0\right) P\left(\vec{x},t \mid \vec{x_0},t_0\right) d\vec{x_0}dt_0 \quad (4)$$

where $P\left(\vec{x},t \mid \vec{x_0},t_0\right)$ is the Probability Density Function (PDF) of the fluid, $\vec{x}$ is the position vector, t is the time, and $S\left(\vec{x_0},t_0\right)$ is the pollutant source terms.

The PDF can be computed following the trajectories of a set of particles in the flow. This set must be big enough to guarantee a certain statistical accuracy. This calculation is made by the Lagrangian particle model LAMBDA [4]. Each particle trajectory can be described by the generalized three-dimensional form of the Langevin equation for velocity

$$\begin{aligned} du_i &= a_i\left(\vec{x},\vec{u},t\right) + b_j\left(\vec{x},\vec{u},t\right) + dW_j(t) \\ dx_i &= \left(U_i + u_i\right)dt \end{aligned} \quad (5)$$

where $i,j = 1,2,3$; U_i is the mean wind velocity vector, $\vec{x}$ is the displacement vector, u_i is the Lagrangian velocity vector, a_i is the deterministic term, b_j is a stochastic term (diffusion coefficient), and the quantity dW_j is a random variable from the incremental Wiener process.

The deterministic (drift) coefficient a_i is computed using a particular solution of the Fokker-Plank equation associated to the Langevin equation. The diffusion coefficient b_j is obtained from the Lagrangian structure function in the inertial subrange. These two coefficients are computed using turbulent velocity variances σ_i^2 and decorrelation time scales $\tau_{L,i}$. These parameters can be obtained employing the Taylor statistical theory on turbulence [5].

At this model, a backward approach has been adopted, where the particles are emitted from the sensor volume and its trajectories are calculated from the time t up to the time t_0, and only those particles which reach the source volume are accounted.

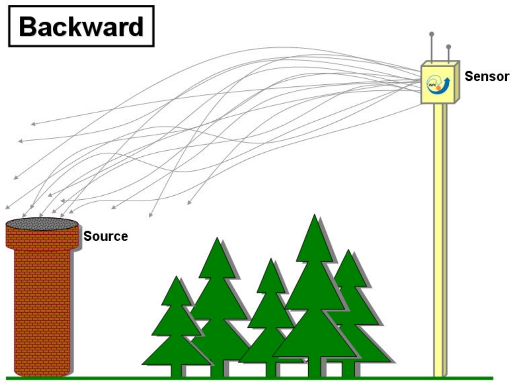

Figure 2. Backward approach.

In order to reduce the CPU-time demanding, this forward problem can be expressed in a source-receptor formulation, where N_f pollutant emitting sources of unknown intensity are related to the pollutant concentration measured in N_s sensors:

$$\vec{C} = M\vec{S} \quad (6)$$

where $\vec{C}$ is a vector representing the mean concentration at the N_s sensors, $\vec{S}$ is a vector representing the intensity on the N_f sources, and M is a state transition matrix computed from the backward model:

$$M_{ij} = \frac{\Delta t}{N_{PES,j}} N_{PVF,i,j} \quad (7)$$

being Δt the time window for the measurements in the sensors, $N_{PES,j}$ is the number of particles emitted by the j^{th} source, and $N_{PVF,i,j}$ is the number of particles emitted by the i^{th} sensor that reached the volume of the j^{th} source.

Inverse problem

Here, we are interested to determine the area source strength using information from an array of sensors. Mathematically, the inverse problems belongs to the class of ill-posed problems. For dealing with this ill-posedness character, a regularization operator is employed [7], associated to the square difference between the experimental data and model generated data. This functional (objective function) can be written as

$$J(S) = \left\| C^{Exp} - MS \right\| + \alpha\Omega(S) \qquad (8)$$

where C^{Exp} is the measured pollutant concentration (experimental data), M is the source-receptor matrix, S is the vector that represents the pollutant source strength, α is the regularization parameter, and $\Omega(.)$ is the regularization operator [6].

RESULTS

The problem consists to identify the emission from an area source. The domain (the ground surface) is divided into sub-domains, and the goal is to determine the emission rate for each sub-domain.

Considering an area with three regions, where each region is emitting with different ratio. The area is divided into 25 sub-domain – see Figure 3. Sub-domains 2 up to 9 are emitting 10 $\mathrm{gm^{-3}s^{-1}}$, and Sub-domains 12 up to 19 the emission ratio is 20 $\mathrm{gm^{-3}s^{-1}}$.

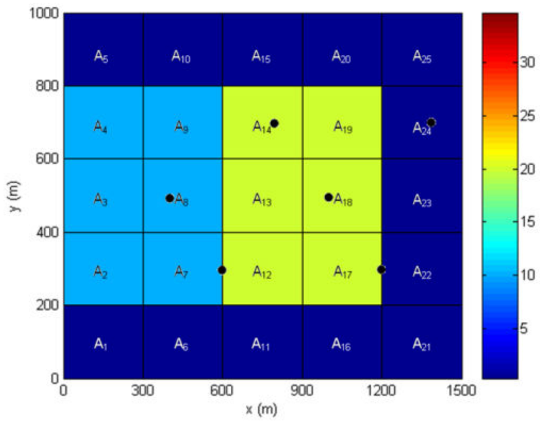

Figure 3. True solution for the area pollutant source. The marks ● are representing the position of the sensors.

Table 1 shows the numerical values for the average from 25 seeds used by the PSO (working with 12 particles and parameters w = 0.2, c_1 = 0.1, and c_2 = 0.2), results obtained from a deterministic technique (quasi-Newton) [8]. The simulation were performed with 5% of noise.

Table 1. Estimation of area source strength by PSO and Q-N algorithms, with addition of 5% noise in experimental data.

Region	Exact	PSO	Error PSO	Q-N	Error Q-N
2	10.00	09.34	0.66	09.82	0.18
3	10.00	10.07	0.07	09.63	0.37
4	10.00	11.26	1.26	11.26	1.26
7	10.00	10.95	0.95	08.76	1.24
8	10.00	10.93	0.93	11.06	1.06
9	10.00	14.99	4.99	15.51	5.51
12	20.00	20.79	0.79	20.12	0.12
13	20.00	19.83	0.17	19.25	0.75
14	20.00	13.06	6.94	11.52	8.48
17	20.00	18.72	1.28	17.88	2.12
18	20.00	22.76	2.76	23.82	3.82
19	20.00	22.47	2.47	23.44	3.44
		Σ	23.27	Σ	28.35

Figure 4 shows the pollutant source emission estimated using the PSO.

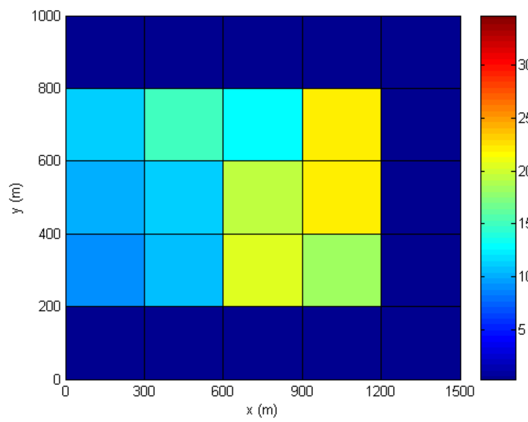

Figure 4. Estimation by PSO with 5% noise.

Table 2 shows the results for similar estimation, but considering 10% of noise in the measured data. The results of the average of 25 seeds are compared with Q-N. Figure 5 displays the estimation obtained using PSO approach.

Table 2. Estimation of area source strength by PSO and Q-N algorithms, with addition of 10% noise in experimental data.

Region	Exact	PSO	Error PSO	Q-N	Error Q-N
2	10.00	09.83	0.17	08.97	1.03
3	10.00	10.40	0.40	09.97	0.03
4	10.00	10.79	0.79	12.52	2.52
7	10.00	10.50	0.50	07.98	2.02
8	10.00	12.06	2.06	10.14	0.14
9	10.00	11.28	1.28	11.56	1.56
12	20.00	14.56	5.44	13.84	6.16
13	20.00	22.67	2.67	22.65	2.65
14	20.00	15.85	4.15	14.14	5.86
17	20.00	21.56	1.56	19.99	0.01
18	20.00	20.05	0.05	21.17	1.17
19	20.00	21.74	1.74	24.90	4.90
		Σ	20.81	Σ	28.05

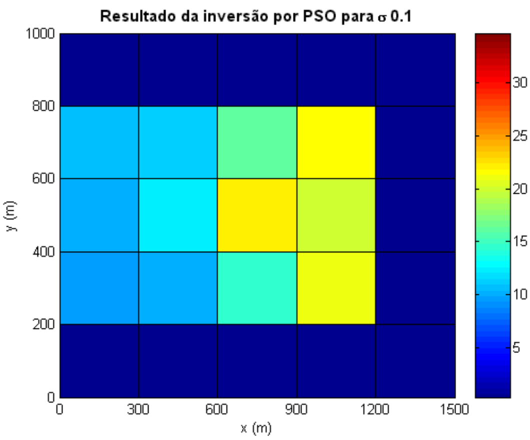

Figure 5. Estimation by PSO with 10% noise.

The PSO technique is about 33.42% faster, in average, as showed in Table 3, to find the solution, proving the promises of this algorithm.

Table 3. Comparison of time between the algorithms.

	Execution Time	Gain
PSO	1748.434s	33.42%
Q-N	2626.597s	-

CONCLUSIONS

The present paper deals with a PSO procedure to determine the emission area sources for the atmospheric pollutant. In order to reduce the computational effort, a source-receptor scheme was employed. Good inverse solutions were obtained using the PSO strategy. Tables 1 and 2 (inversions dealing with 10% and 20% of noise, respectively) show that PSO total error were less than inverse solution computed from the quasi-Newton method.

The main advantage to be pointed out is that in the PSO approach we did not use a regularization operator. The regularization strategy requires the estimation of the regularization parameter, responsible by a fine balance between adhesion term (square difference between measurements and model data) and of the regularization. For the present inverse strategy, it was not necessary.

More tests will be conducted in order to validate another variant of the PSO algorithm. This new variant developed at the Associate Laboratory for Computing and Applied Mathematics introduces the concept of a turbulent atmosphere [9] and has proved to be a valid alternative.

REFERENCES

1. J. Kennedy, R. C. Eberhart, Particle swarm optimization. *Proc. IEEE Int'l. Conf. on Neural Networks*, IV, 1942–1948. Piscataway, NJ: IEEE Service Center, (1995).

2. J. Kennedy, R. C. Eberhart, *Swarm Intelligence*. Morgan Kaufmann. 2001.

3. Shi, Y. and Eberhart, R. C. A modified particle swarm optimizer. Proceedings of the IEEE International Conference on Evolutionary Computation pp. 69-73. IEEE Press, Piscataway, NJ, 1998.

4. E. Ferrero, D. Anfossi. Comparison of PDFs closures schemes and turbulence parametrizations in Lagrangian Stochastic Models, *International Journal on Envorinment and Pollution*, 9, 384(1998).

5. G. A. Degrazia, D. Anfossi, J. C. Carvalho, C. Mangia, T. Tirabassi, H. F. de Campos Velho, Turbulence parametrization for PBL dispersion models in all stability conditions, *Atmospheric Environment*, 34, 3575(2000).

6. H. F. Campos Velho, *Problemas inversos: conceitos básicos e aplicações*. IV Encontro de Modelagem Computacional. Nova Friburgo, RJ. 2003.

7. A. N. Tikhonov, V. Y. Arsenin, *Solution of ill-posed problems*. New York: John Wiley, 1977.

8. D. R. Roberti, *Problemas inversos em física da atmosfera*. Santa Maria: UFSM, 2005. 141 p. Tese (Doutorado em Física) – Centro de Ciências

Naturais e Exatas, Universidade Federal de Santa Maria, 2005.

9. J. C. Becceneri, S. Stephany, H. F. Campos Velho, E. F. P. Luz, Addition of atmosphere turbulence in the Particle Swarm Optimization algorithm. In: *Anais of XXIX CNMAC*. Campinas, 2006.