TWO ALGORITHMS OF THE EXTENDED PSO FAMILY

Juan Luis Fernández-Martínez
Energy Resources Department, Stanford University, Palo Alto, California, U.S.A.
Department of Civil and Environmental Engineering, University of California Berkeley, Berkeley, U.S.A.
Department of Mathematics, University of Oviedo, Oviedo, Spain

Esperanza García-Gonzalo
Department of Mathematics, University of Oviedo, Oviedo, Spain

Keywords: Particle swarm optimization, Cloud algorithms, Stability analysis.

Abstract: In this paper we present two novel algorithms belonging to the extended family of PSO: the PP-GPSO and the RR-GPSO. These algorithms correspond respectively to progressive and regressive discretizations in acceleration and velocity. PP-GPSO has the same velocity update than GPSO, but the velocities used to update the trajectories are delayed one iteration, thus, PP-GPSO acts as a Jacobi system updating positions and velocities at the same time. RR-GPSO is similar to a GPSO with stochastic constriction factor. Both versions have a very different behavior from GPSO and the other family members introduced in the past: CC-GPSO and CP-GPSO. The numerical comparison of all the family members has shown that RR-GPSO has the greatest convergence rate and its good parameter sets can be calculated analytically since they are along a straight line located in the first order stability region. Conversely PP-GPSO is a more explorative version.

1 INTRODUCTION

Particle swarm optimization (PSO) is a global stochastic search algorithm used for optimization motivated by the social behavior of individuals in large groups in nature (Kennedy and Eberhart, 1995). The particle swarm algorithm applied to optimization problems is very simple: individuals, or particles, are represented by vectors whose length is the number of degrees of freedom of the optimization problem. To start, a population of particles is initialized with random positions ($\mathbf{x}_i^0$) and velocities ($\mathbf{v}_i^0$). A same objective function is used to compute the objective value of each particle. As time advances, the position and velocity of each particle is updated as a function of its objective function value and of the objective function values of its neighbors. At time-step $k+1$, the algorithm updates positions $\left(\mathbf{x}_i^{k+1}\right)$ and velocities $\left(\mathbf{v}_i^{k+1}\right)$ of the individuals as follows:

$$\begin{aligned}\mathbf{v}_i^{k+1} &= \omega\mathbf{v}_i^k + \phi_1(\mathbf{g}^k - \mathbf{x}_i^k) + \phi_2(\mathbf{l}_i^k - \mathbf{x}_i^k),\\ \mathbf{x}_i^{k+1} &= \mathbf{x}_i^k + \mathbf{v}_i^{k+1},\end{aligned}$$

with

$$\begin{array}{ll}\phi_1 = r_1 a_g, & \phi_2 = r_2 a_l,\\ r_1, r_2 \in U(0,1) & \omega, a_l, a_g \in \mathbb{R},\end{array}$$

where $\mathbf{l}_i^k$ is the $i-$th particle's best position, $\mathbf{g}^k$ the global best position on the whole swarm, $\phi_1,\ \phi_2$ are the random global and local accelerations, and ω is a real constant called inertia weight. Finally, r_1 and r_2 are random numbers uniformly distributed in $(0,1)$, to weight the global and local acceleration constants, a_g and a_l.

PSO is the particular case for $\Delta t = 1$ of the GPSO algorithm (Fernández-Martínez and García-Gonzalo, 2008):

$$\begin{aligned}v(t+\Delta t) =& (1-(1-\omega)\Delta t)\, v(t)\\ &+ \phi_1 \Delta t\,(g(t) - x(t)) + \phi_2 \Delta t\,(l(t) - x(t)),\\ x(t+\Delta t) =& x(t) + v(t+\Delta t)\Delta t.\end{aligned}$$

This model was derived using a mechanical analogy: a damped mass-spring system with unit mass, damping factor,$1-\omega$ and total stiffness constant, $\phi = \phi_1 + \phi_2$, the so-called PSO continuous model:

$$\begin{cases} x''(t) + (1-\omega)\, x'(t) + \phi x(t) = \phi_1 g(t-t_0) + \phi_2 l(t-t_0),\\ x(0) = x_0,\\ x'(0) = v_0,\\ t \in R.\end{cases} \tag{1}$$

Based on this physical analogy we were able to:

Luis Fernández-Martínez J. and Garcia-Gonzalo E..
TWO ALGORITHMS OF THE EXTENDED PSO FAMILY.
DOI: 10.5220/0003085702370242
In *Proceedings of the International Conference on Evolutionary Computation* (ICEC-2010), pages 237-242
ISBN: 978-989-8425-31-7

1. To analyze the PSO particle's trajectories (Fernández-Martínez et al., 2008) and to explaining the success in achieving convergence of some popular parameters sets found in the literature (Carlisle and Dozier, 2001), (Clerc and Kennedy, 2002), (Trelea, 2003).
2. To generalize PSO to any time step (Fernández-Martínez and García-Gonzalo, 2008), the so-called Generalized Particle Swarm (GPSO). The time step parameter has a physical meaning in the mass-spring analogy but it is really a pseudo-parameter in the optimization scheme that facilitates convergence.
3. To derive a family of PSO-like versions (Fernández-Martínez and García-Gonzalo, 2009), where the acceleration is discretized using a centered scheme and the velocity of the particles can be regressive (GPSO), progressive (CP-GPSO) or centered (CC-GPSO). The consistency of these algorithms have been explained in terms of their respective first and second order stability requirements. Although these regions are linearly isomorphic, CC-GPSO and CP-GPSO are very different from GPSO in terms of convergence rate and exploration capabilities. These algorithms have used to solve inverse problems in environmental geophysics and in reservoir engineering (Fernández-Martínez et al., 2009), (Fernández-Martínez et al., 2010a), (Fernández-Martínez et al., 2010b), (Fernández-Martínez et al., 2010c).
4. To perform full stochastic analysis of the PSO continuous and discrete models (GPSO) (Fernández-Martínez and García-Gonzalo, 2010b), (Fernández-Martínez and García-Gonzalo, 2010a). This analysis served to analyze the GPSO second order trajectories, to show the convergence of GPSO to the continuous PSO model as the discretization time step goes to zero, and to analyze the role of the oscillation center on the first and second order continuous and discrete dynamical systems.

In this contribution, following the same theoretical framework we present two additional developments:

1. We introduce two other novel PSO-like methods: the PP-GPSO and the RR-GPSO (García-Gonzalo and Fernández-Martínez, 2009). These algorithms correspond respectively to progressive and regressive discretizations in acceleration and velocity. PP-GPSO has the same velocity update than GPSO, but the velocities used to update the trajectories are delayed one iteration, thus, PP-GPSO acts as a Jacobi system updating positions and velocities at the same time. RR-GPSO is similar to a GPSO with stochastic constriction factor. Both versions have a very different behavior from GPSO and the other family members introduced in the past: CC-GPSO and CP-GPSO. RR-GPSO seems to have the greatest convergence rate and its good parameter sets can be calculated analytically since they are along a straight line located in the first order stability region. Conversely PP-GPSO seems to be a more explorative version, although the behavior of these algorithms can be partly problem dependent. Both exhibit a very peculiar behavior, very different from other family members, and thus they can be called distant PSO relatives. RR-GPSO seems to have the greatest convergence rate of all of them.
2. We present two different versions of the cloud algorithms: the particle-cloud algorithm and the coordinates algorithm that take adavantages from the idea that GPSO, CC-GPSO and CP-GPSO optimizers are very consistent for a wide class of benchmark functions when the PSO parameters are close to the upper border of the second order stability region. We show also that this situation is slightly different for PP-GPSO and RR-GPSO.

2 THE IMMEDIATE PSO FAMILY

GPSO, CC-GPSO, CP-GPSO correspond to a centered discretization in acceleration and different kind of discretizations in velocity (Fernández-Martínez and García-Gonzalo, 2009). Introducing a $\beta-$discretization in velocity $(\beta \in [0,1])$:

$$x'(t) \simeq \frac{(\beta-1)x(t-\Delta t)+(1-2\beta)x(t)+\beta x(t+\Delta t)}{\Delta t}$$

then, CP-GPSO corresponds to $\beta = 1$ (progressive), CC-GPSO to $\beta = 0.5$ (centered) and GPSO to $\beta = 0$ (regressive). If $\Delta t = 1$ they will be called PSO, CC-PSO and CP-PSO respectively. The β-GPSO algorithm can be written in terms of the absolute position and velocity $(x(t), v(t))$ as follows:

$$\begin{pmatrix} x(t+\Delta t) \\ v(t+\Delta t) \end{pmatrix} = M_\beta \begin{pmatrix} x(t) \\ v(t) \end{pmatrix} + b_\beta,$$

where

$$\begin{aligned} M_{\beta 1,1} &= 1+(\beta-1)\Delta t^2\phi \\ M_{\beta 1,2} &= \Delta t(1+(\beta-1)(1-w)\Delta t) \end{aligned}$$

$$M_{\beta 2,1} = \Delta t\phi \frac{(1-\beta)\beta\Delta t^2\phi - 1}{1+(1-w)\beta\Delta t}$$

$$M_{\beta 2,2} = (1-\beta\Delta t^2\phi)\frac{1+(1-w)(\beta-1)\Delta t}{1+(1-w)\beta\Delta t}$$

and

$$\begin{aligned} b_{\beta 1} = &\ \Delta t^2 (1-\beta)\left(\phi_1 g(t-t_0) + \phi_2 l(t-t_0)\right) \\ b_{\beta 2} = &\ \Delta t \frac{\phi_1 (1-\beta)(1-\beta \Delta t^2 \phi) g(t-t_0) + \beta \phi_1 g(t+\Delta t - t_0)}{1+(1-w)\beta \Delta t} \\ &+ \Delta t \frac{\phi_2 (1-\beta)(1-\beta \Delta t^2 \phi) l(t-t_0) + \phi_2 \beta l(t+\Delta t - t_0)}{1+(1-w)\beta \Delta t} \end{aligned}$$

The first and second order stability regions of the β-GPSO depends on β (Fernández-Martínez and García-Gonzalo, 2009). Figure 1 shows the first and second order stability regions with the associated spectral radii for $\beta = 0.75$ and $\Delta t = 1$ (β-PSO). It is similar to the CP-PSO case $(\beta = 1)$. In fact, when $0 \leq \beta \leq 0.5$, the regions of first and second order stability are single domains, evolving from the GPSO towards the CC-GPSO type, and when $0.5 < \beta \leq 1$ both regions are composed of two zones, evolving towards the CP-GPSO stability regions as β increases. The

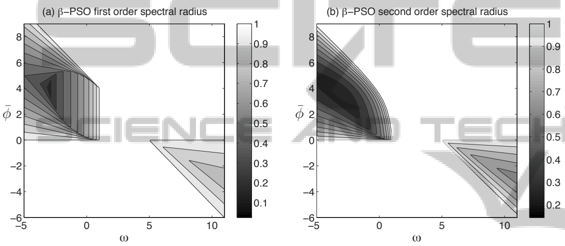

Figure 1: First and second order stability regions for β-PSO $(\beta = 0.75)$ and associated spectral radius.

change of variables to make a β-GPSO version with parameters $a_g,\ a_l, w, \Delta t$, correspond to a standard PSO $(\Delta t = 1)$ with parameters $b_g,\ b_l$, and γ is:

$$b_g = \frac{\Delta t^2}{1+(1-w)\beta \Delta t} a_g$$

$$b_l = \frac{\Delta t^2}{1+(1-w)\beta \Delta t} a_l$$

$$\gamma = \frac{1+(1-w)(\beta-1)\Delta t}{1+(1-w)\beta \Delta t}.$$

Good parameter sets are close to the upper limit of second order stability (Fernández-Martínez and García-Gonzalo, 2009). Figure 2 shows for the Griewank, Rosenbrock, Rastrigin and De Jong-f4 functions the median logarithmic error for 50 dimensions, 100 particles, after 300 iterations and 50 runs for a lattice of $(\omega, \overline{\phi})$ points located on the GPSO first stability region.

2.1 The Extended PSO Family

The PP-GPSO is derived by using progressive discretizations in acceleration and in velocity to approximate the PSO continuous model (1):

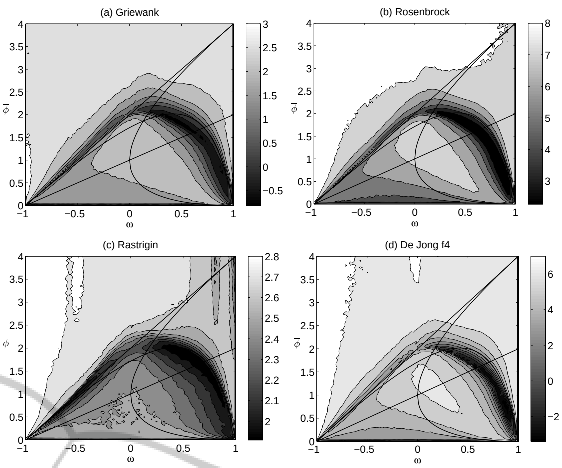

Figure 2: PSO: Mean error contourplot (in $\log_{10}$ scale) for the Griewank, Rosenbrock, Rastrigin and De Jong f4 functions in 50 dimensions.

$$x'(t) \simeq \frac{x(t+\Delta t) - x(t)}{\Delta t}$$

$$x''(t) \simeq \frac{x'(t+\Delta t) - x'(t)}{\Delta t} = \frac{x(t+2\Delta t) - 2x(t+\Delta t) + x(t)}{\Delta t^2},$$

The following relationships apply:

$$x(t+\Delta t) = x(t) + v(t)\Delta t,$$

$$\frac{v(t+\Delta t) - v(t)}{\Delta t} + (1-\omega) v(t) = \\ \phi_1 \left(g(t-t_0) - x(t)\right) + \phi_2 \left(l(t-t_0) - x(t)\right).$$

Adopting $t_0 = 0$ we arrive at:

$$\begin{aligned} v(t+\Delta t) = &\ (1-(1-\omega)\Delta t) v(t) \\ &+ \phi_1 \Delta t \left(g(t) - x(t)\right) + \phi_2 \Delta t \left(l(t) - x(t)\right), \\ x(t+\Delta t) = &\ x(t) + v(t)\Delta t. \end{aligned}$$

which has the same expression for the velocity that the GPSO. The unique difference is that the velocity used to update the trajectory is $v(t)$ instead of $v(t+\Delta t)$ that is used in the GPSO. PP-PSO is the particular case where the time step is $\Delta t = 1$.

First and second order stability region can be deduced using the same methodology that in the other family members (Fernández-Martínez and García-Gonzalo, 2009), that is, writting the first and second order moments as dynamical systems and looking for the region of the $(\omega, \overline{\phi})$ plane where the eigenvalues of the iterative matrix are on the unit circle.

Figure 3 shows the first and second order stability regions of the PP-PSO case $(\Delta t = 1)$ with the associated spectral radii. For the case of second order region the parameter α has been set to 1 in this case. Both regions of stability are bounded. The correspondence

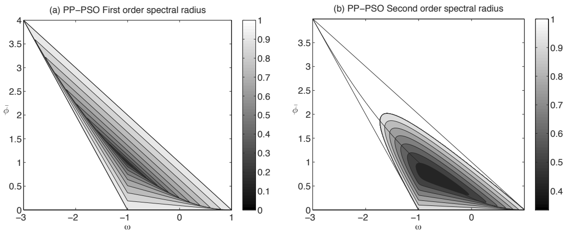

Figure 3: PP-PSO: First and second order stability regions and corresponding spectral radii.

between discrete trajectories for the GPSO and PP-GPSO are:

$$\omega_{PSO} = \omega_{PP} + \Delta t \phi_{PP},$$
$$\phi_{PSO} = \phi_{PP}.$$

Figure 4 shows for the PP-PSO case the logarithmic error for the Griewank, Rosenbrock, Rastrigin and De Jong-f4 case for 50 dimensions, 100 particles, after 300 iterations and 50 runs. Compared to figure 2 it can be observed that PP-PSO provides greater misfits than the PSO, since PP-PSO updates at the same time the velocities and positions of the particles. Also it can be observed that the algorithm does not converge for $\omega < 0$, and the good parameter sets are in the complex region (see figure 4) close to the limit of second order stability and close to $\overline{\phi} = 0$. These results can be partially altered when the velocities are clamped or the time step is decreased.

3 RR-GPSO: REGRESSIVE-REGRESSIVE DISCRETIZATION

The PP-GPSO is derived by using regressive discretizations in acceleration and in velocity to approximate the PSO continuous model (1) :

$$x'(t) \simeq \frac{x(t) - x(t-\Delta t)}{\Delta t}.$$
$$x''(t) \simeq \frac{x'(t) - x'(t-\Delta t)}{\Delta t} = \frac{x(t) - 2x(t-\Delta t) + x(t-2\Delta t)}{\Delta t^2}.$$

The following relationships apply:

$$x(t) = x(t-\Delta t) + v(t)\Delta t,$$

$$\frac{v(t) - v(t-\Delta t)}{\Delta t} + (1-\omega)\, v(t) + \phi(x(t-\Delta t) + v(t)\Delta t) = \phi_1 g(t-t_0) + \phi_2 l(t-t_0),$$

And we can express $v(t)$ as:

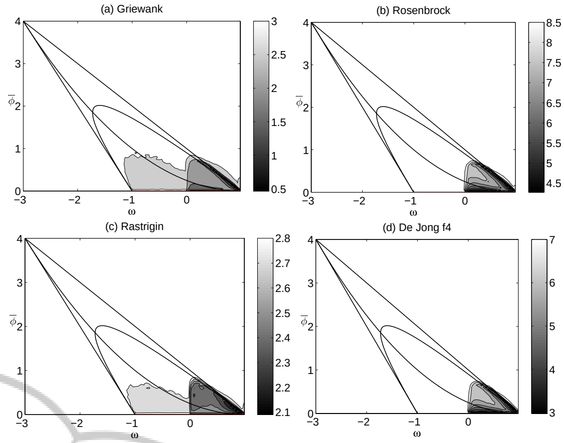

Figure 4: PP-PSO: Mean error contourplot (in $\log_{10}$ scale) for the Griewank, Rosenbrock, Rastrigin and De Jong f4 functions in 50 dimensions.

$$v(t) = \frac{v(t-\Delta t) + \phi_1 \Delta t\,(g(t-t_0) - x(t-\Delta t))}{1 + (1-\omega)\Delta t + \phi \Delta t^2} + \frac{\phi_2 \Delta t\,(l(t-t_0) - x(t-\Delta t))}{1 + (1-\omega)\Delta t + \phi \Delta t^2}.$$

The natural choice for t_0 is Δt. Thus the RR-GPSO algorithm with delay one becomes:

$$\begin{aligned} v(t+\Delta t) &= \frac{v(t) + \phi_1 \Delta t\,(g(t) - x(t)) + \phi_2 \Delta t\,(l(t) - x(t))}{1 + (1-\omega)\Delta t + \phi \Delta t^2} \\ x(t+\Delta t) &= x(t) + v(t+\Delta t)\Delta t,\ t,\ \Delta t \in \mathbb{R} \\ x(0) &= x_0,\ v(0) = v_0. \end{aligned} \tag{2}$$

RR-GPSO with delay one is a particular case of (2) for a unit time step, $\Delta t = 1$. RR-GPSO is a PSO-like algorithm where the parameter

$$A(\omega, \phi, \Delta t) = \frac{1}{1 + (1-\omega)\Delta t + \phi \Delta t^2}$$

could be interpreted as a similar constriction factor to this introduced by Clerc and Kennedy (Clerc and Kennedy, 2002).

Figure 5 shows for $\Delta t = 1$ (RR-PSO case) and $\alpha = 1$ $(a_g = a_l)$, the first and second order stability regions with the corresponding first and second order spectral radii. Both regions of stability are unbounded. Also in both cases the first and second order spectral radii are zero at the infinity: $(\omega, \overline{\phi}) = (-\infty, +\infty)$ and $(\omega, \overline{\phi}) = (+\infty, -\infty)$. The correspondence between discrete trajectories for the GPSO and RR-GPSO are:

$$\omega_{PSO} = \frac{\omega_{RR} - \Delta t \phi_{RR} + (1-\omega_{RR})\Delta t + \Delta t^2 \phi_{RR}}{1 + (1-\omega_{RR})\Delta t + \Delta t^2 \phi_{RR}},$$
$$\phi_{PSO} = \frac{\phi_{RR}}{1 + (1-\omega_{RR})\Delta t + \Delta t^2 \phi_{RR}}.$$

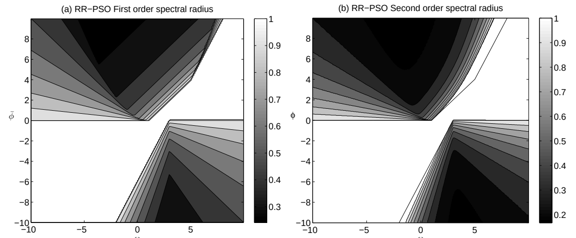

Figure 5: RR-PSO: First and second order stability regions and corresponding spectral radii.

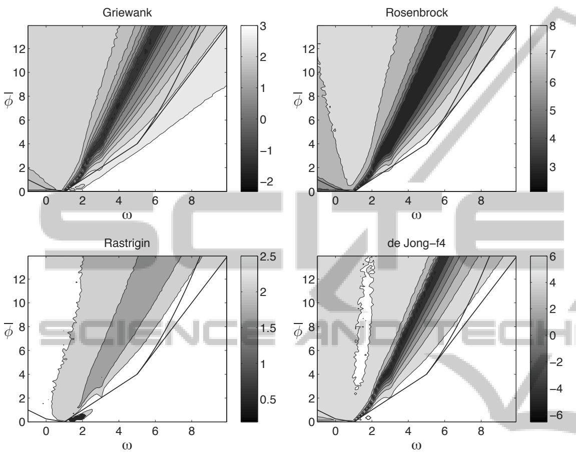

Figure 6: RR-PSO: Mean error contourplot (in $\log_{10}$ scale) for the Griewank, Rosenbrock, Rastrigin and De Jong f4 functions in 50 dimensions.

The good parameters sets for the RR-PSO are concentrated around the line $\overline{\phi} = 3(\omega - 3/2)$, mainly for inertia values greater than two (figure 6). This line is the same for both functions and seems to be invariant when the number of parameters increase. This result is very different from the ones shown for the other version, since the good parameters are not in relation with the second order stability upper border. This line is located in a zone of medium attenuation and high frequency of trajectories. This last property allows for a very efficient and explorative search around the oscillation center of each particle in the swarm.The theoretical and numerical results shown for different PSO family members share the following observations:

1. The PSO algorithms perform fairly well for a very broad region of inertia and total mean acceleration. This region is close for all PSO members to the upper limit of the second order stability region for PSO, CC-PSO, CP-PSO and PP-PSO. For RR-PSO the good points are along a straight line located in a zone of medium attenuation and high frequency of trajectories.
2. These regions are fairly the same for different kind of benchmark functions. This means that the same $(\omega, \overline{\phi})$ points can be used to optimize a wide range of cost functions.

Based on this idea we have designed a PSO algorithm where each particle in the swarm has different inertia (damping) and local and global acceleration (rigidity) constants, being the $(\omega, \overline{\phi})$ sets located in the low misfit regions. This idea has been implemented for the particle-cloud PSO algorithm and extended for CC-PSO and CP-PSO.

The particle-cloud algorithm works as follows:

1. The misfit contours to design the clouds are based on the Rosenbrock function in 50 dimensions. The Rosenbrock function was chosen since in inverse problems the equivalent models that fit the observed data within the same tolerance are located on flat valleys.
2. For each $(\omega, \overline{\phi})$ located on the low misfit region, we generate three different (ω, a_g, a_l) points corresponding to $a_g = a_l$, $a_g = 2a_l$ and $a_l = 2a_g$. Particles are randomly selected depending on the iterations. The algorithm keep track of the (ω, a_g, a_l) points used to achieve the global best solution in each iteration. Thus, when these clouds are used to optimize other benchmark functions with lower complexity, the variability associated to these points might be damped adequately using the time step parameter (Δt).

It was shown that the criteria used to select the points belonging to the cloud it is not very rigid, since points located on the low misfit region (those that close to the second order convergence border) provide very good results, especially those that lie inside the complex zone of the first order stability region. Also, adding some popular parameter sets found in the literature (Carlisle and Dozier, 2001), (Clerc and Kennedy, 2002), (Trelea, 2003) did not improve the results.

Table 1 shows the results obtained for different benchmark functions in 50 dimensions, 100 particles, 300 iterations for 50 runs, using the particle cloud algorithm. The misfits are compared in to the reference values calculated with the program published by Birge (Birge, 2003). It can be observed that the CC-PSO and PSO are the most performing algorithms for all the benchmark functions except for the Rastrigin case. In all the cases the misfits are similar or even better than those presented in the literature. Nevertheless, as pointed before, in inverse modeling it is not only important to achieve very low misfits but also to explore the space of possible solutions. When these algorithms have to be used in explorative form the cloud versions become a very interesting approach, because there is no need to tune the PSO parameters.

Table 1: Comparison between the particle-cloud modalities and the reference misfit values found in the literature (Birge, 2003) for different benchmark functions in 50 dimensions.

Median	**Griewank**	**Rastrigin**	**Rosenbrock**	**Sphere**
Standard PSO	9.8E-03	81	90	6.9E-11
PSO	9.6E-03	92	**86**	8.9E-19
CC-PSO	**7.4E-03**	*99*	90	1.0E-15
CP-PSO	1.8E-02	86	223	2.0E-07
PP-PSO	*1.0E-01*	91	*251*	*8.4E-02*
RR-PSO	1.2E-02	**39**	89	**2.9E-25**

4 CONCLUSIONS

In this paper we present two more different members of the PSO family: the PP-GPSO and the RR-GPSO. Both versions are deduced from the PSO continuous model adopting respectively a progressive and a regressive discretization in velocities and accelerations. Although they are PSO-like versions, PP-GPSO has the same velocity update than GPSO and RR-GPSO has the form of a PSO with constriction factor, its behavior is very different from the PSO case. Particularly the the best parameters sets of the RR-PSO are concentrated along a straight line located in the complex zone of the first order stability region, but are not in direct relation with the upper limit of the second order stability zone. This behavior is very different from others family members including PP-PSO. The numerical comparison between all the members of the PSO family using their corresponding cloud-algorithms has shown that RR-PSO has a very impressive convergence rate while PP-PSO is a more explorative version.

REFERENCES

Birge, B. (2003). PSOt - a particle swarm optimization toolbox for use with Matlab. In *Swarm Intelligence Symposium, 2003. SIS '03. Proceedings of the 2003 IEEE*, pages 182–186.

Carlisle, A. and Dozier, G. (2001). An off-the-shelf PSO. In *Proceedings of the Particle Swarm Optimization Workshop*, pages 1–6, Indianapolis, Indiana, USA.

Clerc, M. and Kennedy, J. (2002). The particle swarm - explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, 6(1):58–73.

Fernández-Martínez, J. L. and García-Gonzalo, E. (2008). The generalized PSO: a new door to PSO evolution. *Journal of Artificial Evolution and Applications*, 2008:1–15.

Fernández-Martínez, J. L. and García-Gonzalo, E. (2009). The PSO family: deduction, stochastic analysis and comparison. *Swarm Intelligence*, 3(4):245–273.

Fernández-Martínez, J. L. and García-Gonzalo, E. (2010a). *Handbook of Swarm Intelligence –Concepts, Principles and Applications*, chapter What makes Particle Swarm Optimization a very interesting and powerful algorithm? Adaptation, Learning and Optimization. Springer.

Fernández-Martínez, J. L. and García-Gonzalo, E. (2010b). Stochastic stability analysis of the linear continuous and discretePSO models. Technical report, Department of Mathematics. University of Oviedo. Submitted to IEEE Transactions on Evolutionary Computation.

Fernández-Martínez, J. L., García-Gonzalo, E., and Fernández-Álvarez, J. (2008). Theoretical analysis of particle swarm trajectories through a mechanical analogy. *International Journal of Computational Intelligence Research*, 4(2):93–104.

Fernández-Martínez, J. L., García-Gonzalo, E., Fernández-Álvarez, J. P., Kuzma, H. A., and Menéndez-Pérez, C. O. (2010a). PSO: A powerful algorithm to solve geophysical inverse problems. application to a 1D-DC resistivity case. *Joanal of Applied Geophysics*, 71(1):13–25.

Fernández-Martínez, J. L., García-Gonzalo, E., and Naudet, V. (2010b). Particle Swarm Optimization applied to the solving and appraisal of the streaming potential inverse problem. *Geophysics*. Accepted for publication.

Fernández-Martínez, J. L., Kuzma, H. A., García-Gonzalo, E., Fernández-Díaz, J. M., Fernández-Álvarez, J., and Menéndez-Pérez, C. O. (2009). Application of global optimization algorithms to a salt water intrusion problem. *Symposium on the Application of Geophysics to Engineering and Environmental Problems*, 22(1):252–260.

Fernández-Martínez, J. L., Mukerji, T., and García-Gonzalo, E. (2010c). Particle Swarm Optimization in high dimensional spaces. In *7th International Conference on Swarm Intelligence (ANTS 2010)*, Brussels, Belgium.

García-Gonzalo, E. and Fernández-Martínez, J. L. (2009). The PP-GPSO and RR-GPSO. Technical report, Department of Mathematics. University of Oviedo. Submitted to IEEE Transactions on Evolutionary Computation.

Kennedy, J. and Eberhart, R. (1995). Particle swarm optimization. In *Proceedings IEEE International Conference on Neural Networks (ICNN '95)*, volume 4, pages 1942–1948, Perth, WA, Australia.

Trelea, I. (2003). The particle swarm optimization algorithm: convergence analysis and parameter selection. *Information Processing Letters*, 85(6):317 – 325.