



**HAL**  
open science

## Validation de la méthode d'optimisation par essaim particulière basée sur un processus de Markov à temps discret (PageRank PSO): Application à la mécanique

Noëlie Di Cesare, Adnan Lahrire, Laurent Christ, Dominique Chamoret,  
Matthieu Domaszewski

### ► To cite this version:

Noëlie Di Cesare, Adnan Lahrire, Laurent Christ, Dominique Chamoret, Matthieu Domaszewski. Validation de la méthode d'optimisation par essaim particulière basée sur un processus de Markov à temps discret (PageRank PSO): Application à la mécanique. 12e Colloque national en calcul des structures, CSMA, May 2015, Giens, France. hal-01515076

**HAL Id: hal-01515076**

**<https://hal.science/hal-01515076v1>**

Submitted on 27 Apr 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Public Domain

# Validation de la méthode d'optimisation par essaim particulaire basée sur un processus de Markov à temps discret (PageRank PSO) : Application à la mécanique

N. Di Cesare\*, A. Lahrire\*, L. Christ\*, D. Chamoret\*, M. Domaszewski\*

\* IRTES-M3M - EA 7274, Université de Technologie de Belfort Montbéliard, 90010 Belfort Cedex, France

**Résumé** - Basée sur les recherches concernant l'algorithme PageRank utilisé par le moteur de recherche Google, une nouvelle version de l'algorithme d'optimisation par essais particuliers est présentée ici. PageRank-PSO (noté PR-PSO) utilise une chaîne de Markov stochastique pour redéfinir en temps réel l'influence des particules au sein même de l'essaim. Dans cet article, cette méthode d'optimisation est appliquée à un benchmark de treillis usuels.

**Mots clés** — Optimisation, PSO, treillis, mécanique

## 1. L'optimisation par essais particuliers

### 1.1 - Généralités concernant la méthode

Depuis des dizaines d'années, les méthodes d'optimisation metaheuristiques ont fait leurs preuves dans le domaine de l'optimisation difficile. Basées sur des phénomènes naturels, ou l'observation de la nature, elles permettent d'obtenir des solutions convenables à des problèmes d'optimisation *np-difficiles*. Parmi elles, la méthode d'optimisation par essais particuliers [1] [2] a fait l'objet de nombreuses recherches ces dernières années (voir par exemple [3] [4] [5] [6] [7] [8]), afin d'améliorer ses performances.

Ainsi, en utilisant l'algorithme d'optimisation par essaim particulaire PSO, des particules sont définies dans un espace de solutions de dimension  $D$ , et chacune d'entre elles correspond à une potentielle solution au problème d'optimisation considéré. Chaque particule n'a accès qu'à un nombre restreint d'informations locales, c'est à dire des renseignements concernant les particules de son *voisinage* (défini par la topologie - statique [3] [9] ou dynamique [10] [11] [12] - de la population), et est dotée de plusieurs renseignements qui la définissent : sa position dans l'espace des solutions, sa vitesse, et la valeur de la fonction objectif pour sa position actuelle.

Le déplacement des particules dans le domaine des solutions est donc réalisé de la manière suivante

$$\begin{aligned} \mathbf{V}_i^k &= \omega \times \mathbf{V}_i^{k-1} + c_1 \times rand_1 \times (\mathbf{P}_{i,best}^k - \mathbf{X}_i^{k-1}) + c_2 \times rand_2 \times (\mathbf{G}_{i,best}^k - \mathbf{X}_i^{k-1}) \\ \mathbf{X}_i^k &= \mathbf{X}_i^{k-1} + \mathbf{V}_i^k \end{aligned} \quad (1)$$

où  $c_1$  et  $c_2$  sont des facteurs d'accélération,  $\omega$  est la masse d'inertie qui contrôle l'influence de la précédente vitesse de la particule sur la suivante,  $rand_1$  et  $rand_2$  sont des nombres aléatoires distribués dans  $[0,1]$ ,  $\mathbf{V}_i^k$  est la vitesse de la particule  $n^o i$  à l'itération  $k$ ,  $\mathbf{P}_{i,best}^k$  est la meilleure performance personnelle de la particule  $i$  trouvée à l'itération  $k$ , et  $\mathbf{G}_{i,best}^k$  est la meilleure performance du voisinage de la particule  $i$  à l'itération  $k$ .

### 1.2 - Développement d'une nouvelle méthode d'optimisation par essais particuliers basée sur une chaîne de Markov stochastique

Une chaîne de Markov à temps discret est un système mathématique qui décrit les transitions d'un état à un autre, donnés dans un espace d'états discret. Cette méthode mathématique est caractérisée par la propriété de Markov : la prédiction du futur n'est pas déterminée ni par l'état présent, ni par les précédents états passés.

Ainsi, une chaîne de Markov peut aussi être décrite par une séquence de graphes orientés, dans lesquels les arêtes sont pondérées par les probabilités de se déplacer de l'état au temps  $n$ , à l'état suivant au temps  $n+1$ . On peut ainsi décrire ce processus plus simplement, grâce à une *matrice de transition de probabilités*. Si la probabilité de passer de l'état  $i$  à l'état  $j$  en un pas de temps est donnée par  $P(j|i) = C_{ij}$ , alors la matrice de connectivité stochastique est donnée par  $\mathbf{C}_{,j}$ , où  $i$  est l'indice des lignes, et  $j$  l'indice des colonnes de la matrice. Comme la somme des probabilités de passer de l'état  $i$  à tous les autres est toujours 1, on a  $\sum_j C_{ij} = 1$ .

On note aussi que la probabilité de passer d'un état à un autre en  $k$  pas de temps discrets est donnée par  $\mathbf{C}^k$ . Ainsi, on peut définir un vecteur de probabilité stationnaire  $\boldsymbol{\pi}$  que l'on peut définir comme le régime stationnaire du processus de Markov à temps discret considéré. Pour une matrice de connectivité stochastique dont tous les termes sont strictement positifs, le vecteur de probabilité stationnaire  $\boldsymbol{\pi}$  est unique et peut être calculé grâce à l'équation (2) suivante :

$$\forall i \quad \lim_{k \rightarrow \infty} (\mathbf{C}^k)_{i,j} = \pi_j \quad (2)$$

Dans l'algorithme PageRank® de Google®, la chaîne de Markov considérée est connue et représente l'ensemble des pages web et des liens entre elles. Les probabilités de transitions entre chacune d'entre elles peuvent être calculées, et le vecteur PageRank, c'est à dire l'état stationnaire du processus de Markov, est ainsi *déduit* [10]. Dans l'algorithme PageRank-PSO présenté ici [11], on considère le graphe orienté défini par *les nœuds que sont les particules et les arêtes que sont les liens d'influence entre elles*. Ces liens sont orientés de telle sorte que l'arête allant de la particule  $i$  à la particule  $j$  définit l'influence de la particule  $i$  sur la particule  $j$ .

On considère ici que l'on connaît l'état stationnaire du processus de Markov, c'est à dire le vecteur PageRank  $\boldsymbol{\pi}$ , et on va en *déduire* les transitions entre les nœuds du graphe orienté considéré, c'est à dire les *influences entre les particules de l'essaim*. Pour ce faire, on considère que le PageRank de chaque particule  $i$  est défini par le succès relatif de cette particule vis à vis de la meilleure de l'essaim  $\mathbf{G}_{best}$ , comme on le voit dans l'équation (3) suivante.

$$\pi_{target}^T(1, i) = \left| \frac{fitness(\mathbf{G}_{best}) \times 100}{fitness(\mathbf{P}_i)} \right| \quad (3)$$

où  $fitness(\mathbf{X})$  est la valeur de la fonction objectif de la particule  $\mathbf{X}$ .

Ainsi, en utilisant l'algorithme décrit dans [11], on peut calculer une potentielle matrice de connectivité stochastique du graphe considéré, tenant compte du fait que les meilleures particules, c'est à dire celles dont la fonction objectif est la plus petite et le PageRank est le plus grand, sont les plus influentes dans l'essaim.

De cette manière, la topologie de la population est évolutive et change à chaque itération, afin de redéfinir les influences de chaque particule sur le reste de l'essaim. L'algorithme PSO a ainsi été redéfini comme suit :

$$\mathbf{V}_i^k = \omega \times \mathbf{V}_i^{k-1} + c_1 \times rand_1 \times (\mathbf{P}_{i,best}^k - \mathbf{X}_i^{k-1}) + c_2 \times rand_2 \times \sum_{j=1}^n C_{ji} \times [\mathbf{P}_{j,best}^k - \mathbf{X}_i^{k-1}] \quad (4)$$

$$\mathbf{X}_i^k = \mathbf{X}_i^{k-1} + \mathbf{V}_i^k$$

Comme nous l'avons vu précédemment, la particule  $i$  est influencée par toutes les particules de l'essaim, et leur influence respective est donnée par les composantes de la  $i^{eme}$  colonne de  $\mathbf{C}$ , c'est à dire la composante  $C_{ji} \forall j$ .

## 2. Application aux treillis mécaniques

Après avoir appliqué et validé cette méthode sur des fonctions benchmark mathématiques classiques [11] (Sphere, Rosenbrock, Ackley, Rastrigin, Griewank, Bohachevsky, Shifted Ackley, Shifted Rastrigin, Shifted Rosenbrock, Shifted sphere, Rotate hyper-ellipsoid function), nous avons cette fois appliqué cette méthode à des treillis mécaniques.

Dans ce cas, la fonction objectif est la masse de la structure, soumise à des contraintes sur les contraintes mécaniques axiales dans les barres et les déplacements des nœuds, et les variables d'optimisation sont les sections des barres. Le problème d'optimisation considéré peut donc être formulé de la manière suivante :

$$\begin{aligned} & \text{minimiser} && f(\mathbf{X}) \\ & \text{Soumis à} && g_i(\mathbf{X}) \leq 0 \quad i = 1, 2, \dots, m \end{aligned} \quad (5)$$

où  $f(\mathbf{X})$  est la masse du treillis, et  $g_i(\mathbf{X})$  représente les  $i$  contraintes d'inégalité. La variable  $\mathbf{X} = (A_1, A_2, \dots, A_n)$  représente une particule PSO, c'est-à-dire une potentielle solution au problème d'optimisation, un vecteur de potentielles valeurs des variables d'optimisation.

Les problèmes d'optimisation traités sont contraints. Les contraintes concernant les plages d'admissibilité des variables d'optimisation ont été gérées grâce à la méthode *harmony memory* présentée dans [13]. Les contraintes concernant les contraintes mécaniques dans les barres et les déplacements des nœuds ont été gérées grâce à une adaptation de la *méthode de pénalisation* présentée dans [13]. La masse d'inertie  $\omega$  a été choisie de manière à décroître linéairement entre 1.2 et 0.4 au cours des itérations. Le nombre maximal d'itérations a été fixé à 1000.

## 3. Résultats et conclusions

Les résultats obtenus sont résumés dans les tableaux (1) suivants. Les résultats peuvent être comparés avec ceux présentés dans l'article [14]. En effet, les mêmes treillis ont été traités, dans les mêmes conditions. Comme on le voit ici, les résultats obtenus sont convaincants puisque meilleurs que ceux donnés dans la littérature, non seulement parce que la valeur de la fonction objectif est plus faible, mais aussi et surtout parce que *toutes* les contraintes d'optimisation sont vérifiées.

## Références

- [1] J. Kennedy et R. Eberhart, «Particle swarm optimization,» *IEEE International conference on neural networks*, vol. 4, pp. 1942-1948, 1995.
- [2] C. Reynolds, «Flocks, herds and schools: A distributed behavioral model,» *Proceedings of the 14th annual conference on computer graphics and interactive techniques*, pp. 25-34, 1987.
- [3] R. Eberhart et J. Kennedy, «A new optimizer using particle swarm theory,» *Proceedings of the sixth international symposium in micro-machine and human science, MHS '95*, pp. 39-43, 1995
- [4] Y. Jiang, T. Hu, C. Huang et X. Wu, «An improved particle swarm optimization algorithm,» *Applied mathematics and computation*, vol. 193, n° 11, pp. 231-239, 2007.
- [5] P. Angeline, «Using selection to improve particle swarm optimization,» *Proceedings of the IEEE international conference on evolutionary computational intelligence*, pp. 84-89, 1998.

Tableaux 1 - Résultats obtenus : (1) Treillis à 22 barres et (2) treillis à 10 barres

Treillis	22 barres			Treillis	10 barres
	Variables (in2)	Cas de chargement		Variables (in2)	Chargement (1)
	(1)	(2)	(3)		
A1	2,9073	2,1465	0,60027	A1	31,1378
A2	1,2738	1,1947	2,6978	A2	0,10012
A3	0,11017	2,0717	0,5969	A3	23,3941
A4	0,11644	1,0714	1,4318	A4	15,6869
A5	2,8741	0,90091	2,1599	A5	0,10049
A6	1,5886	2,2245	2,3347	A6	0,45539
A7	2,0237	1,8172	0,14755	A7	7,5238
Masse (lb)	993,48	849,25	642,0941	A8	20,8423
				A9	20,9179
				A10	0,10249
				Masse (lb)	5065,84

- [6] A. Mohais, R. Mendes, C. Ward et C. Posthoff, «Neighborhood re-structuring in particle swarm optimization,» *Proceedings of the 18th Australian joint conference on artificial intelligence*, pp. 776-785, 2005.
- [7] P. Suganthan, «Particle Swarm optimizer with neighborhood operator,» *Proceedings of the 1999 congress on evolutionary computation, CEC99*, vol. 3, p. 1962, 1999.
- [8] S. Pasupuleti et R. Battiti, «The Gregarious particle swarm optimizer,» *Proceedings of the 8th annual conference on genetic and evolutionary computation, GECCO' 06*, pp. 67-74, 2006.
- [9] R. Mendes, J. Kennedy et J. Neves, «Watch the neighbor, or how can the swarm learn from its environment,» *Proceedings of the 2003 IEEE swarm intelligence symposium SIS'03*, pp. 88-94, 2003.
- [10] S. Janson et M. Middendorf, «A hierarchical particle swarm optimizer,» *Proceedings of the 2003 congress on Evolutionary computation*, vol. 2, pp. 770 - 776, 2003.
- [11] S. Janson et M. Middendorf, «A hierarchical particle swarm optimizer for dynamic optimization problems,» *Lecture notes in computer science - Applications of evolutionary computing*, vol. 3005, pp. 513 - 524, 2004.
- [12] P. Suganthan, «Particle swarm optimizer with neighborhood operator,» *Proceedings of the 1999 congress on evolutionary computation*, vol. 3, p. 1962, 1999.
- [13] A. Langville et C. Meyer, *Google's PageRank and beyond. The science of search engine rankings*, Princeton university press, 2012.
- [14] N. Di Cesare, D. Chamoret et M. Domaszewski, «A novel hybrid PSO algorithm based on a stochastic Markov chain model,» *Soumis à : Advances in Engineering software*, 2014.
- [15] P. Newton, J. Mason, K. Bethel, L. Bazhenova, J. Nieva et P. Kuhn, «A stochastic Markov chain model to describe lung cancer growth and metastasis,» *PLoS ONE*, vol. 7, n° 14, p. e34637, 2012.
- [16] Y. Xu, M. Domaszewski, D. Chamoret, W. Zhang et J. Korvink, «Particle swarm optimization for non-convex problems of size and shape optimization of trusses,» *Proceedings of the eleventh international conference on computational structures technology*, paper 67, 2012.
- [17] R. Asl, M. Aslani et M. Shariat Panahi, «Sizing optimization of truss structures using a hybridized genetic algorithm,» *Cornell university library*, 2013.