

# Back to random topology

Maurice Clerc

27th March 2007

## 1 Introduction

In 2004 I defined and put on line a simple PSO with random topology. More precisely the communication topology (infornetwork) was randomly modified after an iteration if there had been no improvement of the global best.

However, although the C code is freely available from now several years, and although it has been published in 2005 in my book [2] with some explanations (and in 2006 in the English translation [3]), it appears that some people not understood well the method that is used. Hence this short paper, in which I simply explain how I found it, starting from my first bad idea to a more effective one.

## 2 Method 1: the very first (and bad) idea

Let  $\mathcal{S} = \{P_1, \dots, P_S\}$  be the swarm, where each  $P_i$  is a particle. In classical local best PSO, each particle is informed by  $K$  particles, including itself. Usual values are swarm size  $S = 20$ , and neighbourhood size  $K = 3$ . Note that by “neighbourhood” I actually mean “informants”, so that there is no confusion: it is easy to understand “a particle informs itself”, as saying “it is a neighbour of itself” sounds a bit strange.

Now, if we want to define a random neighbourhood for particle  $P_i$ , it seems natural to choose at random  $K$  particles in  $\mathcal{S}$  (according to an uniform distribution). However, doing just that, we are not sure  $P_i$  is informed by itself, which seems quite ridiculous. Moreover, it is well known that a fixed topology in which each particle does not inform itself is not very effective. So it does not seem to be a good idea even for a random topology.

So, it is better to define the random set of informants of  $P_i$  thanks to two rules:

- it contains  $P_i$
- the  $K - 1$  other particles are chosen at random in  $S$ .

Why not in  $S - \{P_i\}$ ? Because it seems experimentally that it is better that a particle may sometimes have no informant at all, except itself. So it can perform

Nb of informants	Probability
1	0.0025
2	0.1425
3	0.855

Table 1: Probability distribution of the number of informants, for  $S = 20$  and  $K = 3$

local search around its best known position. By choosing in  $S$ , this situation occurs with a probability equal to  $1/S^{K-1}$ . By choosing in  $S - \{P_i\}$ , of course it never happens.

It is clear that the maximum number of informants (i.e. neighbours and the particle itself) is equal to  $K$ , but, more precisely, we can define the probability distribution of the number of informants, as shown on Table 1, for  $S = 20$  and  $K = 3$ . The general formula is given in Annexe 6.1.

This distribution is not satisfying. On the one hand I want a mean value around 3 (here 2.85), for I know it is better for multimodal problems. But on the other hand I also want that *sometimes* the particle may have a lot of informants. In other words I want a distribution in which all values between 1 and  $S$  have a non null probability, so that all possible topologies may appear, between “all particles are independent” and “each particle is informed by the whole swarm”, *but* with a small mean value. It is impossible with this method, as we can see for example on Figure 1 where  $K = S$ : in order to obtain a non null probability for big values, the mean has also to be quite big.

### 3 Method 2

Of course, it is always possible to directly manipulate the distribution, but such a method needs some additional parameters. There is a simpler way: instead to choose at random who informs who, I can choose at random who is informed by whom. The final result is a nice distribution of the sum of  $S - 1$  independent binary Bernoulli variables, as shown on Figure 2 (see the exact formula in Annexe 6.2). Note that the mean value is now greater than  $K$  (2.9 for  $K = 2$ , 3.71 for  $K = 3$ ). And although it is not visible on the figure, the probability is never equal to zero.

Let us explain how it is done in practise:

- step 1 - we built a  $S \times S$  matrix  $L$ . We immediately set  $L(i, i) = 1$ , i.e. the whole diagonal (each particle informs itself)
- step 2 - for each line  $i$  we draw at random (uniform distribution on  $\{1, \dots, S\}$ )  $K$  numbers  $k_1, \dots, k_K$ , and we set  $L(i, k_n) = 1$ . Of course it may happen that the same element is set to 1 several times

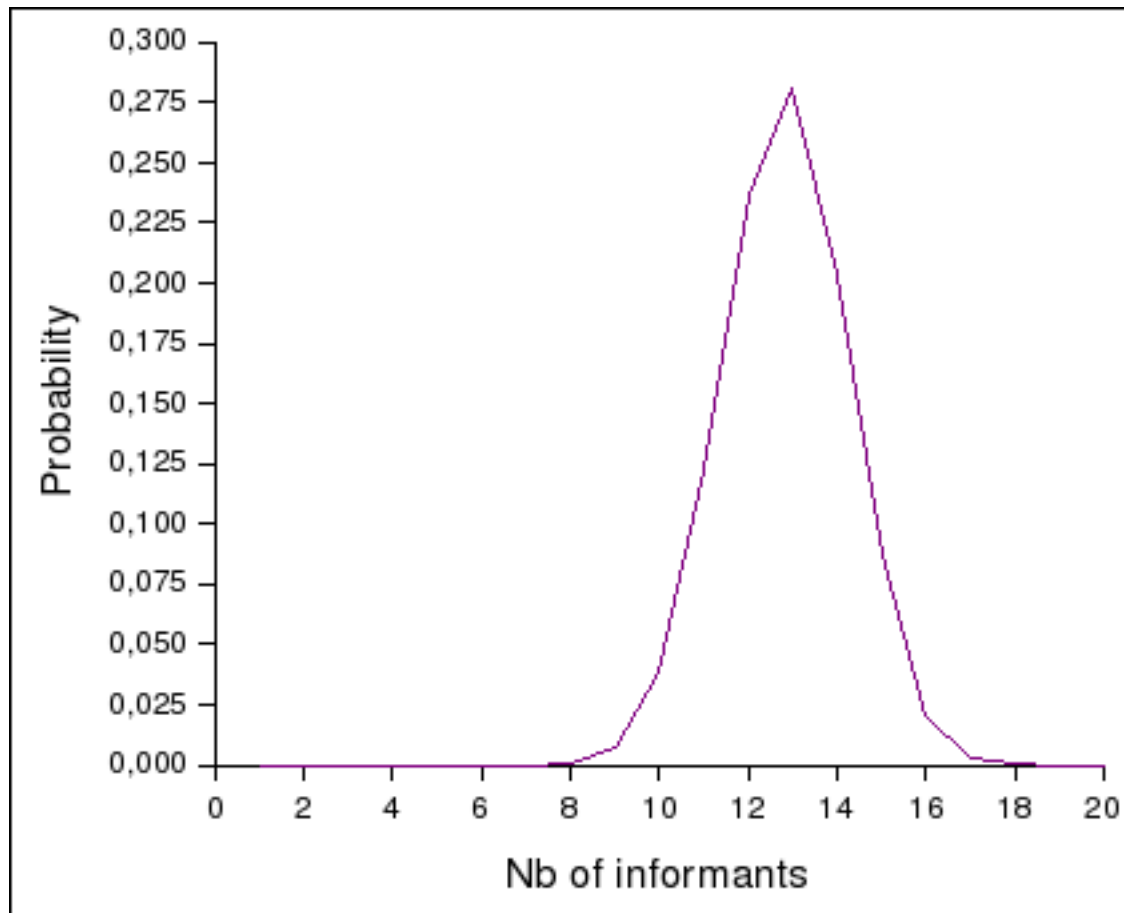


Figure 1: Probability distribution of the number of informants with Method 1, and  $K = S$

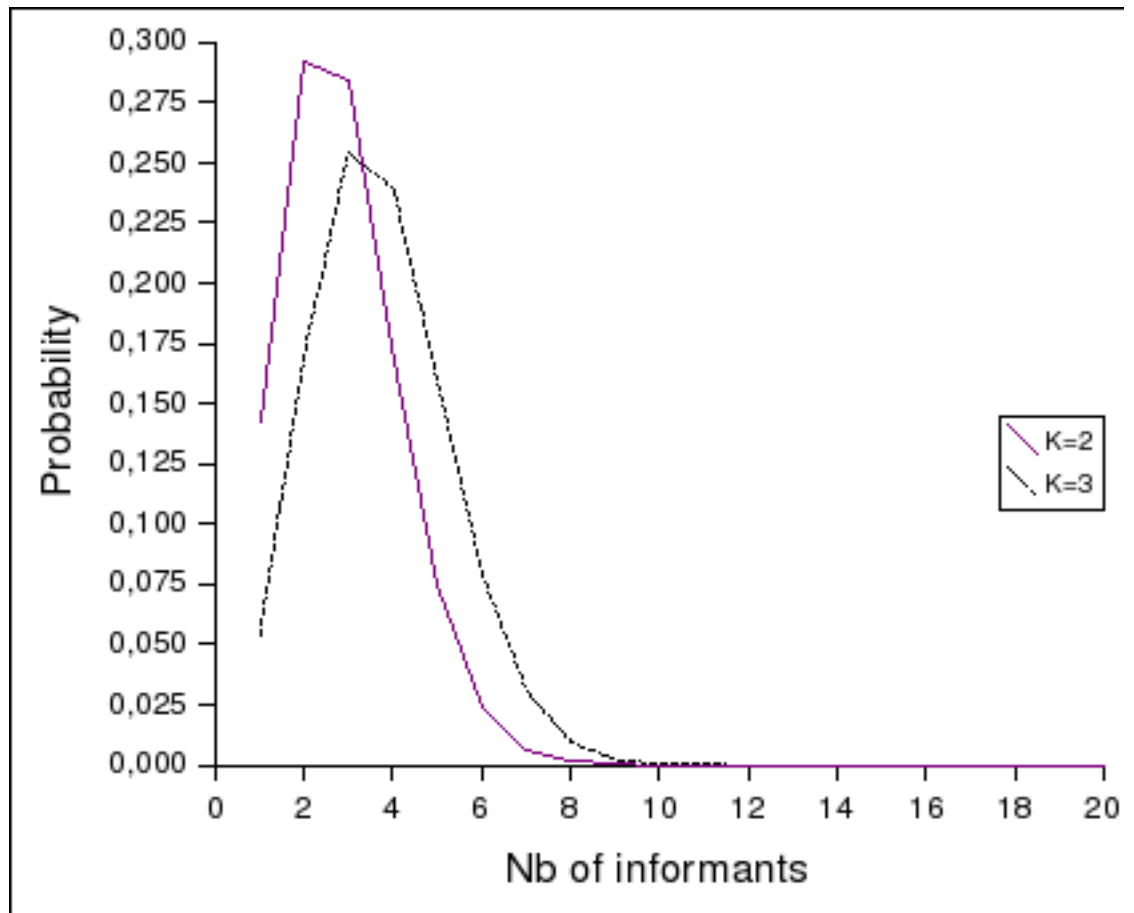


Figure 2: Probability distribution of the number of informants with Method 2, and  $K = S$

- step 3 - then we consider for each particle  $j$  the column  $L(., j)$ . If  $L(i, j) = 1$ , it means that particle  $P_i$  informs particle  $P_j$ . Note that for Method 1 we consider in fact for each particle  $j$  the *line*  $L(j, .)$ .

It is interesting to note that this method is formally equivalent to a more recent one, called “Stochastic Star” [5] (see Annexe 6.3).

## 4 Method 3

Although we do have with Method 2 a complete range of possible neighbourhood sizes, the probability to have a big one is very small. Also we can take this opportunity of using a random topology to try to remove the arbitrary parameter  $K$ .

In the above description of the Method 2, in Step 2 we can replace the constant  $K$  by a random one. After all it would be just an application of the usual rule of thumb “If you don’t know, flip a coin”.

What happens, for example, if we choose it at random uniformly in  $\{1, \dots, S\}$ , i.e.  $K = \mathcal{N}(1, S)$ ? And with a “triangular distribution”, i.e.  $K = \mathcal{T}(1, S) = (\mathcal{N}(1, S) + \mathcal{N}(1, S)) / 2$ ? The exact distribution formulas for the number of informants are quite complicated, and it is easier to simply perform some simulations. As we can see on Figure 3 the results are very similar, and not that good: the mean value is far too high.

This is nevertheless an interesting result: using *too much* randomness is not necessarily a good idea, for the final distribution is just quite similar to a Gaussian one. So Method 3 is not a good one, and there is still room for an improvement of Method 2 (i.e. with a bit higher probabilities for big neighbourhood size values).

## 5 When to modify the topology?

### 5.1 Criterion 1: no improvement

If the best know solution has not been improved after a whole iteration (i.e. after  $S$  fitness evaluations), then the infonetwork is redesigned. Note that the combination Method 2 + Criterion 1 is the one that is used in the basic PSO OEP 0 [2], and later in Standard PSO 2006 (available on the Particle Swarm Central [7]).

### 5.2 Criterion 2: enough time

There is a theoretically better way, based on a rumour spread model. In short any information found by a particle should have time to reach any other one. Clearly it is depending on the diameter of the graph of the current infonetwork. However exactly computing it is quite time consuming. An approximation can

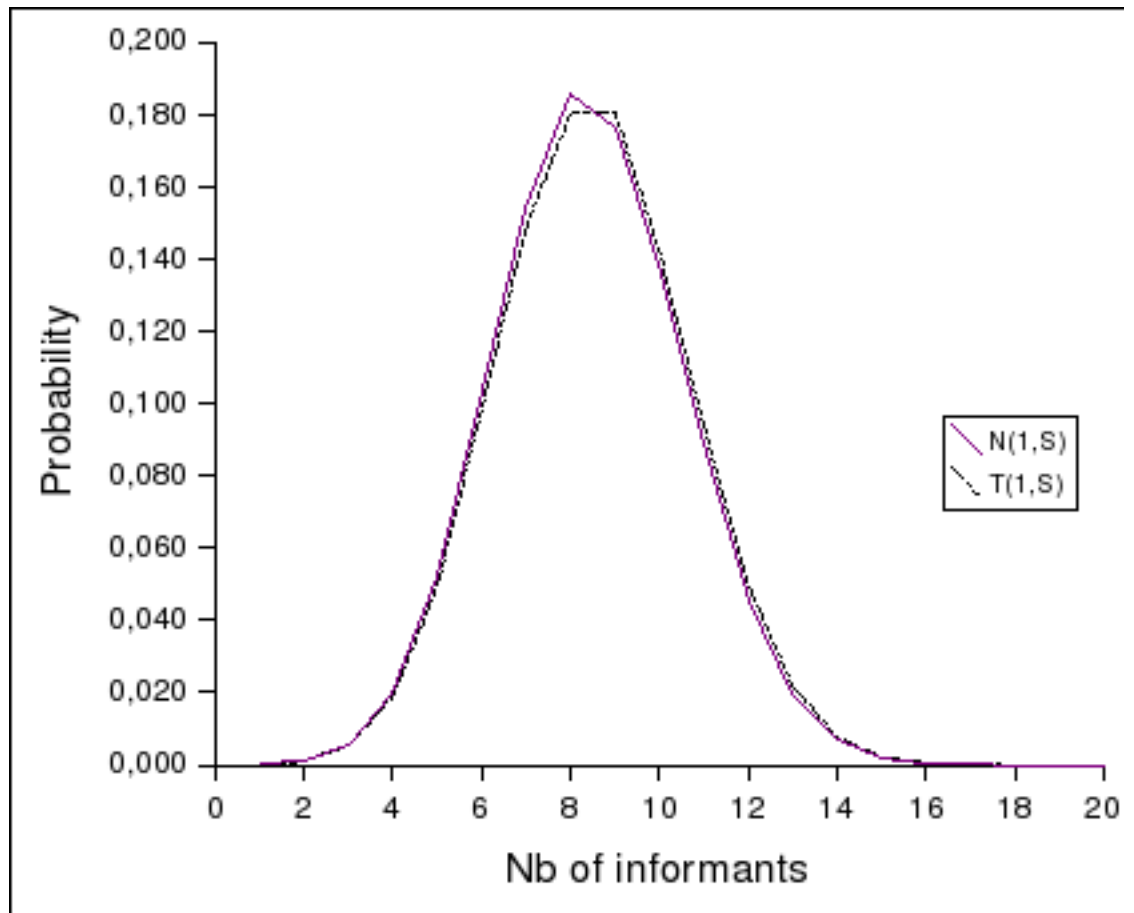


Figure 3: Probability distribution of the number of informants with Method 2, and a random  $K$

be used: if the number of links is  $N$  then the network is redesigned after  $N/2$  iterations.

Actually this method is used in the parameter free PSO called TRIBES [1, 2, 3] although that in this precise algorithm redesigning is not done at random.

## 6 Annexe

### 6.1 Distribution formula for Method 1

When drawing  $K$  times a number between 1 and  $S$  the total number of possible sequences that contains  $n$  different numbers is given by

$$N(S, K, n) = \frac{S!}{S^K (S-n)!} \sum_{i=1}^n (-1)^{n+i} i^K \frac{n!}{i!(n-i)!}$$

or equal to 0 in the three following particular cases

$$\begin{cases} n = 0 \\ n > K \\ n > S \end{cases}$$

This can be found by using the recurrence relations

$$\begin{cases} N(S, 1, 1) = 1 \\ N(S, K, n) = nN(S, K-1, n) + (S-n+1)N(S, K-1, n-1) \end{cases}$$

However there is a small difficulty here, for a particle informs itself. If we consider particle 1, for example, it means the first element of all “favorables” sequences must be equal to 1. And also the length of each sequence is not  $K$  but  $K-1$ . So we consider now three kinds of sequences of  $K-1$  values:

- the ones that does not contain 1, and with  $n-1$  different values. Their number is  $N_1 = N(S-1, K-1, n-1)$
- the ones that contain  $n$  different values. Their number is  $N_2 = N(S, K-1, n)$
- the ones that contain  $n$  different values, but *not* 1. Their number is  $N_3 = N(S-1, K-1, n)$

The total number of “favorable” sequences is then  $N_F = N_1 + N_2 - N_3$ . As the total number of sequences is  $S^{K-1}$ , the distribution is given by  $N_F/S^{K-1}$ . For  $K=3$  we have the following table 2. For biggest values, the easiest way is to write a small program.

Number of informants	Probability
1	$\frac{1}{S^2}$
2	$3 \frac{S-1}{S^2}$
3	$\frac{(S-1)(S-2)}{S^2}$

Table 2: Distribution of probability for Method 1,  $K = 3$

## 6.2 Distribution formula for Method 2

Let us consider for example the first particle  $P_1$ , i.e. the first column of the matrix  $L$ . By definition, the probability to have 1 in line 1 is 1 (the particle informs itself). For the  $S - 1$  other lines, the probability is

$$p = 1 - \left(1 - \frac{1}{S}\right)^K$$

So the random variable “number of informants of  $P_1$ ” is simply

$$Y = 1 + \sum_1^{S-1} X_j \quad (1)$$

where  $X_j$  is a Bernouilli random variable of parameter  $p$ . Its distribution is then given by adapting the classical formula, and we find

$$\text{prob}(Y = n) = C_{S-1}^{n-1} \left(\frac{K}{S}\right)^{n-1} \left(1 - \frac{K}{S}\right)^{S-n} \quad (2)$$

## 6.3 Formal equivalence between Method 2 and Stochastic Star

In [5] the “external” informants of a given particle  $P_i$  are chosen according to a probability law. Each other particle  $P_j$  informs  $P_i$  with a probability  $p$ , and the particle informs itself with a probability 1. The authors call this method “Stochastic Star”. In practice, a random number  $r$  is drawn from the uniform distribution  $\mathcal{N}(0, 1)$ . If  $r < p$  then  $P_j$  indeed informs  $P_i$ . If we code “ $P_j$  informs  $P_i$ ” by 1, and the contrary by 0, we can see that the random variable “number of informants” is exactly the one defined by Equation 1. So its probability distribution is the same than for Method 2 when

$$p = 1 - \left(1 - \frac{1}{S}\right)^K \quad (3)$$

On the one hand this probability threshold  $p$  can be directly set to any value, so the method is a bit more flexible. However, on the other hand, with formula ?? we better see why and how  $p$  is depending on  $S$  and  $K$ . For example, as we know  $S = 20$ ,  $K = 3$  is quite effective for multimodal problems, it means that



for such problems we should choose, with the Stochastic Star approach,  $S = 20$ ,  $p = 0.14$ .

Note that though *the way* information is used by a given particle is different in, say, Standard PSO 2006 and in[5]. In Standard PSO 2006 only the best informant is kept, as in[5] the velocity update equation contains a weighted combination of all informants, for it is a kind of FIPS (Fully Informed Particle Swarm) [4, 6].

## References

- [1] Maurice Clerc. TRIBES - Un exemple d'optimisation par essaim particulière sans paramètres de contrôle. In *OEP'03 (Optimisation par Essaim Particulaire)*, Paris, 2003. 14 pages.
- [2] Maurice Clerc. *L'optimisation par essaims particuliers. Versions paramétriques et adaptatives*. Hermès Science, 2005.
- [3] Maurice Clerc. *Particle Swarm Optimization*. ISTE (International Scientific and Technical Encyclopedia), 2006.
- [4] Rui Mendes. *Population Topologies and Their Influence in Particle Swarm Performance*. PhD thesis, Universidade do Minho, 2004.
- [5] Vladimiro Miranda and N. W. Oo. New experiments with epso - evolutionary particle swarm optimization. In *IEEE Symposium on Swarm Optimization*, Indianapolis, USA, 2006.
- [6] Rui Ward Christopher Postoff Christian Mohais, Arvind Mendes. Neighborhood re-structuring in particle swarm optimization. In Ray Jarvis Shichao Zhang, editor, *18th Australian Joint Conference on Artificial Intelligence (AI 2005)*, pages 776–785, Sydney, Australia, December 2005. Springer-Verlag.
- [7] PSC. Particle Swarm Central, <http://www.particleswarm.info>.