In this paper, we consider sensors that move according to the well-known Particle Swarm Optimization (PSO) scheme in order to improve network coverage. Unlike the original PSO, particle speed is updated by considering a consensus algorithm based on local optimum position. Two different versions of the algorithm have been simulated: a global version that allows nodes to use information of the whole sensor field and a local version based only on neighborhood information. The algorithm based on global information is used as comparison term for the local version. Also, a variant of these algorithms has been implemented by adding the concept of pioneers, which are powerful sensors that explore the field to detect interesting areas before the other sensors become active. In order to evaluate the performance of our schemes, different scenarios have been introduced by varying the probability areas for events to occur in. The performance of the network has been evaluated in terms of coverage and energy consumption for movement and has shown that the proposed techniques obtain remarkable results for both parameters considered.

Categories and Subject Descriptors
C.2 [Computer-Communication Networks]: Network Architecture and Design; Network Topology; C.2.3 [Network Operations]: Network Management

Keywords
Wireless Sensor Networks, Particle Swarm Optimization, Consensus Algorithm, Swarm Intelligence

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

1. INTRODUCTION

The social insect metaphor for solving problems has become a hot topic in the last years. This approach emphasizes distributedness, direct or indirect interactions among simple agents, flexibility and robustness. Swarm Intelligence is inspired to the swarms of animals such as ants, bees, birds, etc. The single animal typically has limited capabilities, but when it reaches a group, the group is able to solve complex problems such as the nest building, or the path search from a food source to the nest. Many research activities have been conducted to design metaheuristics based on the behaviors of natural swarms. Concerning a sub-area of the Swarm Intelligence, the Particle Swarm Optimization (PSO) can be considered as an optimization stochastic technique based on flocks of birds and groups of fishes. In the PSO, a set of software agents, called particles, search for the best solution of a specific continuous optimization problem. In this paper, we focus on a modified PSO scheme in order to solve coverage problems in Wireless Sensor Networks (WSNs). Sensors collect data related to the events occurred in the sensor field. The movements of the sensors aim to improve the coverage and minimize the energy consumption. The speed update for each sensor is based also on a consensus mechanism, well-known in the multi-agents schemes. In order to evaluate the performance of our techniques, we conducted extensive simulations. Specifically, simulation results showed that the techniques considered allow to monitor specific area with a good quality of coverage. Furthermore, based on the results of the simulation we added the concept of pioneers, which are more powerful sensors, able to discover interesting areas and to drive the other simple sensors to reach these specific zones. The contribution of this work can be summarized as follows:

- it proposes a scheme for event coverage maximization that is distributed, reactive and scalable. On the contrary of all the existing schemes for coverage in WSN based on PSO, which are centralized and/or need global information in order to allow nodes to update their velocity, our proposal includes a variant that is completely distributed and requires only local information. The local information needed by each node...
is the number of sensed events per unit of time by itself and its neighbors. If the spatial distribution of events changes in the time, nodes will react by moving to different areas, richer of events than the currently covered area. Finally, an increase of the number of nodes has a very moderate effect on the performance of the scheme, and its behavior does not change when the number of nodes increases beyond a certain value.

- The modification introduced to the classical PSO scheme is accomplished by the introduction of consensus techniques in the decision process of the nodes. The introduction of consensus algorithms gives to the nodes the capability to take decisions in a dynamic, localized and reactive fashion.
- When all nodes of the network move to locations where they can sense more events, the overall energy consumption of the network can become an issue. Hence, this work introduces the usage of pioneer sensors that explore the sensor field in order to determine the areas richer of events and effectively communicate this information to all other nodes causing their movement.
- It extensively simulates and evaluates the performance of the distributed algorithm with and without the pioneers, in comparison with its centralized version and in terms of coverage accuracy and energy consumption.

The rest of the paper is organized as follows: in Section 2 we review the related works. In Section 3 we describe PSO techniques and Consensus algorithms and then we introduce the proposed solution schemes. In Section 4 we illustrate the system model and give the details about the simulations. In Section 5 we present and discuss the results of the proposed schemes and finally we conclude our paper in Section 6.

2. RELATED WORKS

Wireless sensor networks (WSNs) have been successfully adopted in many strategic applications such as target tracking, surveillance and classification. Coverage and target detection probability are the two most significant factors for the performance of WSNs, which are determined by dynamic deployment algorithms based on a given optimization criterion [9]. Recently, mobile sensor networks have been under investigation and it has been shown that mobility, while complicating the design of higher layer algorithms, can also improve the network, for instance, in terms of capacity, coverage, lifetime, etc. [8].

Most of PSO applications in communications have been focused on clustering in ad-hoc networks aiming to minimize energy consumption. In [11] PSO has been used for distance based clustering of wireless sensor networks.

An interesting summary of the main recent results in literature regarding the PSO in WSNs, is given in [14]. Authors summarize the PSO algorithms by classifying them according to: task to accomplish, optimization criterion, scheme used (i.e. centralized or distributed) and type of work (i.e. simulation or real deployment). Three works [1], [2], [3] are cited as algorithms proposed for supporting the deployment of mobile nodes in order to maximize the coverage of the sensor network. Specifically in [1], PSO is used to deploy the sensors in the field, while the evaluation of the coverage is done by using the Voronoi Tessellation.

In order to evaluate the fitness function, they considered not only the vertex of the various polygons, but also those of n points randomly selected along the edges of polygons. Authors of [2] consider the maximization of the coverage as an optimization criterion by implementing a centralized technique As their technique is based on a modified PSO strategy they called their technique Particle Swarm Genetic Optimization (PSGO).

In [3], authors propose a dynamic deployment algorithm named Virtual force directed co-evolutionary PSO (VFCPSO), where they combine the co-evolutionary PSO (CPSO) with the Virtual Force algorithm, exploiting the advantages of both the techniques. The CPSO algorithm requires global information in order to make nodes update their velocities. Three more algorithms have been proposed that are not cited in [14]: two of them are based on PSO through the usage of Voronoi Tessellation [4], [17] and the third proposes a deployment strategy based on global information that exploits parallel PSO (PPSO) to reach an effective coverage [21].

All the cited PSO algorithms for coverage have in common the usage of a centralized scheme and/or the need of global information to solve the problem by using PSO. Under this respect, it is important to observe that in WSNs distributed approaches are preferred in respect of centralized solutions even though a distributed strategy is more difficult to be designed. The technique proposed in this work is of distributed type; thus, in what follows, we focus our attention on the scientific contributions that consider distributed approaches. Important results in this sense have been given in [15], [16], where the localization of wireless sensor nodes is handled. Applications for sensor networks based on consensus include load balancing [7], multi-vehicle control and navigation [16], flocking [18]. Much of the existing work is focused on average-consensus problems The study of the recent literature allows us to conclude that the formulation of WSNs issues as multidimensional optimization problems and through bio-inspired technique is a research field in full swing. The issue faced by the proposed technique is the deployment of wireless sensor nodes through a distributed approach to obtain a better coverage of regions of interest.

To the best of our knowledge there is no a similar technique in literature that works for the same task and directly comparable to our solution strategy and for this reason we will use as term of comparison a version based on global information.

3. PROPOSED ALGORITHM

In this section, we will give some relevant details about PSO and Consensus Algorithms and then we will introduce our scheme based on the modification of PSO by the usage of an explicit consensus mechanism.

3.1 Particle Swarm Optimization

PSO is a very versatile population-based swarm intelligence technique [5]. The particles of PSO are located within a research space and evaluate a fitness function based on their position. They can move around the research space by combining their history with the information received by one or more neighbors in the swarm. If we assume that the particles move in a 2-D space, in PSO the new velocity of the particle i will be calculated as:
\[ x_i(t + 1) = x_i(t) + v_i(t + 1) \]

The three terms of the velocity update equation (1) characterize the particles’ behavior: the first term is called inertia (or moment) because it keeps track of the particle’s previous velocity, the second term is the cognitive component and is used to make particles keep track of their best known positions, the third term is the social component because it represents the best position achieved by the swarm.

### 3.2 Consensus Algorithms

The study of information flow and interaction among multiple agents in a group plays an important role in understanding the coordinated movements of these agents. As a result, a critical problem for coordinated control is to design appropriate protocols and algorithms such that the group of agents can reach consensus on the shared information in the presence of limited and unreliable exchange and dynamically changing interaction topologies. The term “consensus” in multi-agent systems indicates the process of reaching an agreement on a certain quantity of interest that depends on the state of all agents. A consensus algorithm states the information exchange between an agent and all of its neighbors in the network [16]. A basic variant of the PSO algorithm works by having a population of candidate solutions called particles. In our scenario the particles are the sensor devices that exchange information and move to reach a common objective. In [20], Vicsek proposes a consensus algorithm based on a time discrete model of n autonomous agents that move in a research space. The state of each agent is updated by a rule based on its state and the state of its neighbors, where its neighbors are all the agents located in a limited transmitting/receiving range. The evolution of the system can be described as follows:

\[ x_i(t + 1) = x_i(t) + u_i(t) \]

where:

\[ u_i = (w_i - 1) \cdot x_i(t) + \sum_{j \in N_i(t)} w_{ij} \cdot x_j(t) \]

In (3) and (4), \( x_i \) is the status of agent \( i \), \( u_i \) is the control law on agent \( i \), \( N_i \) is the set of neighbors of node \( i \) and \( w_{ij} \) is the weight associated to the contribution of the \( j^{th} \) agent on the \( i^{th} \) neighbor. A neighbor is a 1 hop distance node. The strategy of consensus consists in choosing the right weights in order for all the agents to converge to the same state. It is worth to notice that a global consensus is reached thanks to the nodes at the boundaries, that make consensus regions physically connected.

### 3.3 Modified PSO

We are convinced that the PSO scheme is a good solution for sensors network coverage problems, above all for the applications where the energy cost for moving nodes is not too high. In our simulations we used the velocity update equation in (1) by modifying the meaning of the social term in order to determine a placement of nodes that reflects the events probability distribution in a sensor field. Usually, in the PSO scheme, the social component is the best position achieved globally by the swarm in the research space. For our matters, it is not useful to consider a global best position, because it implies a centralized scheme of control or, at least, the capacity of the nodes to communicate with every other node in the sensor field. In order to take into account the limited communication capabilities of sensors, we stated that the social term involves the position that enjoys the maximum consensus within each node’s neighborhood, where a neighborhood is composed only of the sensors within its transmitting/receiving range. Thus, we assume that at each iteration of the modified PSO algorithm, the sensors exchange information and determine the maximum consensus in their neighborhood. The velocity update equation is modified as follows:

\[ v_i(t + 1) = \omega \cdot v_i(t) + \phi_p \cdot r_p \odot (p_i - x_i(t)) + \phi_g \cdot r_g \odot (g_i - x_i(t)) \]

where:

\[ I_i = \frac{\left\| x_k - x_i \right\|}{d_{exp}} \]

In (6), \( x_k \) is the position of the particle in the set of neighbors of \( i \) that obtained the best value of objective function in the previous iteration, and \( d_{exp} \) is a coefficient of repulsion that is meant to avoid sensors’ coverage area overlapping. The computation of the objective function is based on the number of events per unit of time that occurred in the sensor field, by assuming that the nodes can have a global or a local knowledge of the sensor field. It is worthy to note that we used the maximum consensus because only the node that achieves the highest value of objective function is considered in the velocity update equation. This corresponds to set \( w_{ij} = \begin{cases} 1, & j = i, k \\ 0, & \text{elsewhere} \end{cases} \). we left the analysis of the algorithm behavior in the cases of average consensus, obtained by varying the assigned weights \( w_{ij} \), for future developments.

### 4. SYSTEM MODEL

For the analysis of the modified PSO algorithm, we consider a square sensor field \( S \) where a certain number of mobile sensors \( N \) is deployed randomly or uniformly. Events happen in this limited and measurable space according to a probability density function \( \psi(s) \), where \( s \) is the surface unit over \( S \). In this work, we will show only the results when we assume a probability function that does not change during the simulation time.

In order to assess the performance of our scheme, we need to introduce the “ideal” behavior of a coverage algorithm. A straightforward idea is to distribute the sensors present in...
the sensor field according to the events probability function. The fraction of nodes \( n_i \) that should cover a region \( R_i \) of \( S \) can be computed by the following ratio of surface integrals:

\[
n_i = \frac{\int_{R_i} \psi(s) \, ds}{\int_S \psi(s) \, ds}.
\]

In order to give also a visual effect of the behavior of the proposed scheme, we decided to split the sensor field in four regions, characterized by homogeneous probability for an event to occur. In Fig. 1 we show the four scenarios used for the performance evaluation, while in Table 1 we present the parameters used both in the simulations and for the PSO algorithm. Scenarios shown in Figure 1 allow us to test and stress our techniques under “extreme” situations (i.e. Figures 1 (b) and (d)), and in “typical” situations, like Figure 1 (a).

![Figure 1: Simulated scenarios: (a) \( \psi_{A,D} = 0, \psi_{B,C} = 0.5 \), (b) \( \psi_A = 1, \psi_{B,C,D} = 0 \), (c) \( \psi_{A,C} = 0.2, \psi_{B,D} = 0.3 \), (d) \( \psi_{A,B,C,D} = 0.25 \).](image)

<table>
<thead>
<tr>
<th>Table 1: Parameters Used in the Simulations</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Simulation Parameters</strong></td>
</tr>
<tr>
<td>Sensor Field ((L \times L))</td>
</tr>
<tr>
<td>Number of areas ((A))</td>
</tr>
<tr>
<td>Number of scenarios ((N))</td>
</tr>
<tr>
<td>Number of sensors ((N))</td>
</tr>
<tr>
<td>Sensing range ((r_s))</td>
</tr>
<tr>
<td>Communication range ((r_c))</td>
</tr>
<tr>
<td>Repulsion coefficient ((d_{rep}))</td>
</tr>
<tr>
<td>Traveled distance for convergence ((d_{Nk}))</td>
</tr>
<tr>
<td>Number of runs for each scenario</td>
</tr>
<tr>
<td>Confidence interval</td>
</tr>
</tbody>
</table>

| Algorithm Parameters | | |
|----------------------|-----------------|
| Inertial Weight (\(\omega\)) | [0.7±1] |
| Personal Coefficient (\(\phi_p\)) | [0.15±2.1] |
| Global Coefficient (\(\phi_g\)) | [1.5±5] |

As we said in the previous section, the objective function is evaluated by the nodes depending on their global or local knowledge of the sensor field. Specifically, we consider two versions of the objective function calculated at each iteration by each node:

- **GOF**: takes into consideration the position of the node in respect of the events that occurred in the whole sensor field. Hence, this version needs global information and it is called Global Objective Function.
- **LOF**: takes into consideration only the events that occurred within the node’s sensing area. In this case, each node can calculate the objective function depending only on local information and it is called Local Objective Function.

We decided to show the results also of the version that uses global information in order to give a comparison term to the local and distributed algorithm, since the latter can not be compared with any other algorithm because, to the best of our knowledge, no distributed algorithm using PSO for coverage purpose exists in literature, hence a direct comparison would not be fair. The convergence condition we set is that all the swarms have to travel less than \( d_{Nk} \) in the same iteration in order for the algorithm to consider the reached positions as an equilibrium position. This condition can be easily implemented in a distributed manner.

For the evaluation of the results, we will consider the difference between the ideal number of sensors in each area \((n_i, N)\) and the number of sensors located in the same area, when the convergence condition of the algorithm is met, as performance parameter. Since sensors can have overlapping coverage areas, a finer performance parameter should consider not only their number in each area but also their effective coverage of the area. We left this refinement for future developments and introduced the parameter \( d_{rep} \) in order to limit the overlap of sensors. Furthermore, since we are using mobile sensors is also important to quantify the energy spent on the movement. Energy for the movement has been set to a proportional cost model that depends on traveled distance \( d \), as follows:

\[
E_m(d) = kd
\]

where the value of the constant \( k \) can vary between 0.1 and 1. According to [19], we decided to set \( k \) equal to 0.1 \( \frac{1}{m} \).

In this context we neglected the energy consumption related the gossiping, since it is out of scope, but in future work it could be interesting consider it.

### 5. PERFORMANCE EVALUATION

The performance evaluation is based on the four scenarios and the two objective functions introduced in the previous section that have been simulated for both random and uniform initial sensors deployment. In this section, we first show some preliminary results used to determine the values of some important simulation parameters of Table 1, then we move on to illustrate the behavior of the proposed scheme with both the GOF and LOF, and finally, we implement a new version of the proposed scheme that uses *pioneers*. All the results of this section have been averaged on 100 runs with a confidence interval of 95%.

#### 5.1 Parametric Analysis

First of all we perform some preliminaries simulations in order to understand the best value of some important simulation parameters, such as: sensors number and sensing
range. The simulations are performed by using both the objective function as follows:

- by varying the number of nodes in the range \([30 \div 90]\) with a step of 20 nodes;
- by varying the sensing range in the range \([10 \div 40]\) with a step of 10 m.

All other parameters are taken from Table 1 and the scenario considered is (a) of Fig. 1.

From Fig. 2 and Fig. 3 we can see the behavior of the two objective functions, when the initial deployment is random and the number of sensors varies. Specifically, Fig. 2 shows us that the performance of the GOF algorithm is not affected by an increase of the number of nodes, and that for 50 nodes it obtains a minimum value of nodes not correctly positioned. Also, the performance of the LOF keep the same when the number of nodes is larger than 70, and it also shows a smaller value of nodes not correctly positioned for the configuration with 30 and 50 nodes. Instead, the average energy spent for movement per node increases by increasing the number of nodes (Fig. 3). For this reason, even if a different scenario could have needed a different optimal number of sensors, we decided to use 50 nodes as number of sensors present in the sensor field.

As expected, from Fig. 4, we see that a larger sensing range (that implies also a larger communication range), improves the performance in terms of nodes not correctly positioned, but at the same time this means more space to sense and more movement to perform (Fig. 5). Thus, also considered the size of the simulated sensor field, we decided to use 12.5 m as sensing range.

5.2 Global and Local Objective Functions

From an initial uniform distribution of the sensors in the sensor field, Fig. 6 and Fig. 7 present the final positions of the nodes for the Scenario (a) of Fig. 1, after sensors execute the algorithm that uses global or local knowledge of the sensor field, respectively.

Both the objective functions obtain good results, even though the possibility to possess global knowledge is more effective on the final placement of the sensors. These observations are further confirmed by the diagrams of Fig. 8, where the average number of sensors, that are in a wrong area when the algorithm converges, is shown. We can observe that the initial deployment has its importance for the coverage problem, because for all the scenarios and for both the objective functions, the initial uniform deployment always obtains better results in respect of the random deployment. Some experiments with a larger number of areas (>16) showed us that the undesired effect for the nodes to concentrate around some specific points, as in Fig. 7, is not present. Thus, we can expect that our algorithm performs better when a more realistic (i.e., with a very large number of areas) probability distribution of the events in the field.
Secondly, the scheme with GOF performs better than the correspondent scheme with LOF, as we could expect by using global information instead of local one. But, we can also conclude that when the events probability distribution is not completely unbalanced among the areas (as in Scenario (b)), the maximum fraction of sensor in a wrong position is equal to 20%.

In Fig. 9 we illustrate the energy spent for the movement of nodes in each scenario. Thus, we can see that the better performance in terms of coverage obtained by the initial uniform deployment is paid as larger amount of energy spent to move toward the right positions. Furthermore, if we exclude the Scenario (b) (where we already showed that algorithm with LOF is ineffective), the results obtained by the algorithm with LOF needs on average 15% more energy than those obtained from the algorithm with GOF. In one case (Scenario (a) with uniform initial sensor deployment) the energy needed by GOF is more than that used by LOF. In fact, in this case the local exchange of information, starting from a uniform situation, is more effective than the global because the latter makes nodes, especially those in the center of the sensor field, move back and forth among the two events regions. We also considered the time expenditure for convergence in terms of number of iterations, but we have not reported the results because they are very similar to those related to the energy spent. In Fig. 8 and Fig. 9 we evaluated the performance parameters by considering average values of the weights in the fitness function, that is, the various weights were not the “optimal” for every scenario, whereas in Fig. 10 and Fig. 11 we show the results obtained when the best weights are selected for each scenario in an independently fashion. In this case we are able to appreciate better performance both in terms of nodes not correctly positioned and energy expenditure.

From Fig. 8, it is evident that the percentage of nodes not correctly positioned in the worst case, that is the Random LOF, is more than 40%. Once a better weights choice for this algorithm is considered, the percentage decreases to ≈ 30%. Concerning the last scenario, that is the Scenario (d), we do not appreciate substantial differences. In fact, the total number of nodes not correctly positioned is smaller for
Figure 11: Average amount of energy spent to move nodes when the best weights are chosen for every scenario.

every technique, but the difference is between 2.5−5%. Concerning the Scenario (c), we have a significant improvement higher than 10% and some improvements can be also noticed for the Scenario (a) (∼6% for Random GOF). Moreover, it is worth to note the differences in terms of energy consumption when the different weights are considered. In some cases the energy consumption is drastically reduced. For instance, we can observe a reduction of 12 unit of energy concerning the technique Uniform GOF (Scenario (d)), or 13.5 unit of difference as far as the Uniform GOF (Scenario (a)) is concerned. In summary, we appreciate a global improvement both in terms of nodes not correctly positioned and energy consumption based on different choices of the weights.

5.3 Pioneer Sensors

In the previous sections we analyzed the proposed modified PSO algorithm with GOF and LOF by considering only one type of sensor. In practice, all the sensors present in the sensor field had the same capacities, size, sensing range, etc. The results of the performance suggest that this solution is expensive in terms of energy spent for move the nodes. For this reason we added a new type of sensor that we defined pioneers. We imagine pioneers as more powerful in terms of energy and sensing capabilities, thus we set their sensing range to 50 m and we do not consider the energy spent in their movements. Their aim is to explore the sensor field in order to determine interesting areas in terms of occurred events. Once their objective is reached, they spread out the information along the sensor field and activate the normal sensors. For our simulation we used a number of pioneers equal to the 20% of the total number of sensors, and consider a certain number of iteration (10% of the total number) for the pioneers to explore the sensor field. The concept of pioneers is general and could be also considered in other contexts.

From Fig. 12 we can compare the results about the number of nodes in a wrong area with the same plot of Fig. 8. The obtained results, for this quality parameter, are very close to those obtained without using pioneers above all for the algorithm with GOF and for the most “balanced” scenarios ((a), (c) and (d) of Fig. 1). However, the remarkable results are given by the energy spent for movement. In Fig. 13 we can see that the average amount of energy spent by “normal” node to reach a final position that reflects the events probability distribution is much lower, between 50% and 80% of saved energy. In order to complete the performance analysis, we consider the total energy spent both by the “normal” nodes and the pioneers nodes. In Figure 14 we can observe that the even the total energy spent from both the normal nodes and pioneers nodes is taken into account, the network behaves better in terms of energy consumption than in the case with all homogeneous nodes.

Figure 12: Percentage of nodes not correctly positioned when pioneers are used.

Figure 13: Average amount of energy spent by the nodes for movement when pioneers are used.

Figure 14: Average amount of energy spent by both “normal” nodes and pioneers nodes for movement.

6. CONCLUSION

In this paper we focused on the synergetic action of the PSO and the WSNs. Specifically, we implemented some PSO techniques by considering the sensors as agents, dis-
distributed in a virtual searching area. Our devices are able to collect information about events that occur in the sensor field and they move depending on a specific update formula, based on neighborhood information averaged by a consensus scheme. We evaluated the behavior of the algorithm in terms of achieved coverage and energy consumption. Simulation results showed that nodes moving with our scheme are able to cover the interesting areas. Furthermore, we tried to outperform the techniques considered in terms of energy consumption by adding the concept of pioneers. Through the usage of these specific sensors we were able to obtain a remarkable coverage of the interesting zones and drastically reduce the energy consumption. In this work we considered a time-varying metric in order to qualitatively measure the capability of nodes to react to changes in the events distribution will be the subject of future investigation.

Acknowledgment
This work has been carried out under the framework of STEM-Net, PRIN-National Italian Project #H21J11000050001, financed by the Italian Ministry of University and Research.

7. REFERENCES