Particle Swarm Optimization in Wireless Sensor Networks: A Brief Survey

Raghavendra V. Kulkarni, *Senior Member, IEEE,* and Ganesh Kumar Venayagamoorthy, *Senior Member, IEEE*

***Abstract*—Wireless sensor networks (WSNs) are networks of autonomous nodes used for monitoring an environment. Developers of WSNs face challenges that arise from communication link failures, memory and computational constraints, and limited energy. Many issues in WSNs are formulated as multidimensional optimization problems, and approached through bio-inspired techniques. Particle swarm optimization (PSO) is a simple, effective and computationally efficient optimization algorithm. It has been applied to address WSN issues such as optimal deployment, node localization, clustering and data-aggregation. This paper outlines issues in WSNs, introduces PSO and discusses its suitability for WSN applications. It also presents a brief survey of how PSO is tailored to address these issues.**

***Index Terms*—clustering, data-aggregation, localization, optimal deployment, PSO, Wireless sensor networks**

I. Introduction

Wireless sensor networks (WSNs) are an emerging technology [1] that has potential applications in surveillance, environment and habitat monitoring, structural monitoring, healthcare, and disaster management [2]. A WSN monitors an environment by sensing its physical properties. It is a network of tiny, inexpensive autonomous nodes that can acquire, process and transmit sensory data over wireless medium. One or more powerful base stations serve as the final destination of the data. The the properties of WSNs that pose technical challenges include dense ad-hoc deployment, dynamic topology, spatial distribution and constrains in bandwidth, memory, computational resources and energy.

WSN issues such as node deployment, localization, energy-aware clustering and data-aggregation are often formulated as optimization problems. Traditional analytical optimization techniques require enormous computational efforts, which grow exponentially as the problem size increases. An optimization method that requires moderate memory and computational resources and yet produces good results is desirable, especially for implementation on an individual sensor node. Bio-inspired optimization methods are computationally efficient alternatives to analytical methods. Particle swarm optimization (PSO) is a popular multidimensional optimization technique [3]. Ease of implementation, high quality of solutions, computational efficiency and speed of convergence are strengths of PSO. Literature is replete with applications of PSO in WSNs. The objective of this paper is to give a flavor of PSO to researchers in WSN, and to give a qualitative treatment of optimization problems in WSNs to PSO researchers in order to promote PSO in WSN applications.

The rest of this paper is organized as follows: PSO and its relative advantages are briefly outlined in Section II. Sections III through VI discuss applications of PSO in optimal deployment, localization, clustering and data-aggregation (also referred to as data-fusion). In each of these sections, a specific WSN issue is introduced, and a brief description of how PSO is applied to address the particular issue is presented. Finally, a projection of future PSO applications in WSNs and concluding remarks are given in Section VII.

II. PSO: A Brief Overview

A. The PSO Algorithm

PSO models social behavior of a flock of birds [3]. It consists of a swarm of s candidate solutions called particles, which explore an n-dimensional hyperspace in search of the global solution (n represents the number of optimal parameters to be determined). A particle i occupies position X_{id} and velocity V_{id} in the d^{th} dimension of the hyperspace, $1 \leq i \leq s$ and $1 \leq d \leq n$. Each particle is evaluated through an objective function $f(x_1, x_2, \cdots, x_n)$, where $f : \mathbb{R}^n \rightarrow \mathbb{R}$. The cost (fitness) of a particle close to the global solution is lower (higher) than that of a particle that is farther. PSO thrives to minimize (maximize) the cost (fitness) function. In the global-best version of PSO, the position where the particle i has its lowest cost is stored as ($pbest_{id}$). Besides, $gbest_d$, the position of the best particle. In each iteration k, velocity V and position X are updated using (1) and (2). The update process is iteratively repeated until either an acceptable $gbest$ is achieved or a fixed number of iterations $k_{\max}$ is reached.

$$\begin{aligned} V_{id}(k+1) &= w \cdot V_{id}(k) + \varphi_1 \cdot r_1(k) \cdot (pbest_{id} - X_{id}) \\ &\quad + \varphi_2 \cdot r_2(k) \cdot (gbest_d - X_{id}) \qquad (1) \\ X_{id}(k+1) &= X_{id}(k) + V_{id}(k+1) \qquad (2) \end{aligned}$$

Here, φ_1 and φ_2 are constants, and $r_1(k)$ and $r_2(k)$ are random numbers uniformly distributed in [0,1]. This is the basic "textbook" information about PSO. Popular themes of PSO research are: choice of parameters and their ranges, iterative adaption of parameters, particle interaction topologies, convergence acceleration, adaption to discrete,

Authors acknowledge the support received for this work from the National Science Foundation, USA, under the grants ECCS # 0529292 and ECCE # 0348221.

Authors are with the Real-Time Power and Intelligent Systems Laboratory, Missouri University of Science and Technology, Rolla, MO-65409, USA e-mail: ({arvie,gkumar}@ieee.org).

Manuscript received October 21,2009; revised March 02, 2010.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMCC.2010.2054080

binary and integer domains, and hybridization with other algorithms. The state-of-the art in PSO is presented in [4].

B. Other Optimization Algorithms

Traditional optimization methods include linear, nonlinear and quadratic programming, Newton-based techniques and interior point methods. Their computational complexities grow exponentially with the problem size. Resource requirements and cost of mathematical programming engines (such as IBM ILOG CPLEX) used for linear, nonlinear and quadratic programming make them unattractive for resource constrained nodes. This is the motivation for heuristic algorithms such as PSO, genetic algorithm (GA), differential evolution (DE) and bacterial foraging algorithm (BFA). GA facilitates evolution of the population generation by generation using operators such as crossover, mutation and selection [5]. DE is similar to GA, but it uses a differential operator [6], which creates a new solution vector by mutating an existing one by a difference of randomly chosen vectors. BFA models the foraging behavior of bacteria that use a combination of straight line and random movements to reach nutrient rich locations [7]. Advantages of PSO over these alternatives are:

1) Ease of implementation on hardware or software.
2) Availability of guidelines for choosing its parameters.
3) High quality solutions because of its ability to escape from local optima [8], [9].
4) Availability of variants for real, integer and binary domains [4].
5) Quick convergence [10], [11].

PSO with s number of n-dimensional particles that runs for $k_{\max}$ iterations requires $k_{\max} \cdot s$ fitness evaluations and memory for $s \cdot n$ variables each for positions, velocities, and $pbest$, plus n variables for $gbest$. This can be prohibitively expensive on some nodes.

III. Optimal WSN Deployment

WSN deployment problem refers to determining positions for sensor nodes (or base stations) such that the desired coverage, connectivity and energy efficiency can be achieved with as few nodes as possible [12]. Events in an area devoid of an adequate number of senor nodes remain unnoticed; and the areas having dense sensor populations suffer from congestions and delays. Optimally deployed WSN assures adequate quality of service, long network life and financial economy. Available PSO solutions to the deployment problem are computed centrally on a base station for determining positions of sensors, mobile nodes or base stations as summarized in Table I.

A. Stationary Node Positioning

Objective of the centralized, off-line *PSO-Voronoi* algorithm proposed by Aziz *et al* in [13] is to minimize the area of coverage holes. The strategy is based on the principle that if each point in the region-of-interest (ROI) is covered by a sensor, then the whole ROI is covered. Assessment of coverage involves sampling the ROI through grid scan. *PSO-Voronoi* circumvents this by Voronoi polygons around the sensors. PSO particles are sensors positions. For each particle, a set of Voronoi polygons are determined, and the vertexes of the polygons are treated as sample points. The cost function is the number of vertexes that are uncovered by sensors. *PSO-Voronoi* achieves close to ideal coverage but ignores the time complexity of determining Voronoi polygons.

Hu *et al*. have proposed *PSO-Traffic* for topological planning for a real world traffic surveillance application [14]. The study uses a large number of camera loaded nodes, some of which require larger transmission radii facilitated by expensive high-power transmitters. The objective is to determine the nodes with high power transmitters such that the highest possible connectivity is achieved at the lowest possible hardware expense. *PSO-Traffic* is binary PSO in which the particles represent sequences of sensors. PSO seeks to minimize a multi-objective fitness parameter $LDC = a \cdot L + b \cdot D + c \cdot C$, where L is the transmission hop of the signal, D is the increase in conflict and C is the cost of the extra high power transmitters. Constants a, b and c define the relative weights of L, D and C respectively. L and D are computed from the scaled length and the scaled degree, concepts from the small world phenomenon. This algorithm has resulted in symmetric distribution of high power transmitters, improved network performance and a saving in system cost.

B. Mobile Node Positioning

Li *et al*. have proposed a mixture of stationary and mobile nodes and particle swarm genetic optimization (*PSGO)* as a remedy to coverage holes [15]. The *PSGO* hybrid is employed to determine redeployment positions of mobile nodes in order to improve average node density. *PSGO* maximizes quality-of-service, defined as the ratio of the area covered to the total area of the ROI, $QoS = S_c/S$, which should be ideally equal to unity. The area covered S_c is $S_c = S_{nod_c} \cup S_{rob_c}$, the union of the area covered by the stationary nodes and the robot-assisted mobile nodes. S_c only depends on the sensing radius r_s and the positions (x and y coordinates) of the N mobile nodes, $S_c = f(x_{rob_1} \cdots x_{rob_N}, y_{rob_1} \cdots y_{rob_N}, r_s)$, which *PSGO* determines. *PSGO* borrows the mutation and selection operations from GA. In each iteration, *PSGO* discards some worst particles and generates an equal number of new particles at random locations. Besides, it moves a few particles randomly. The paper reports as high as 6% increase in QoS with 5 out of 100 static nodes replaced by mobile nodes. Mobile nodes can be repositioned using*PSGO*dynamically as the network topology changes. But, it necessitates mechanisms for obstacle avoidance and location awareness.

VFCPSO: Wang *et al*. have proposed a virtual force co-evolutionary PSO (*VFCPSO*) for dynamic deployment of nodes for enhanced coverage in [16]. Virtual force based dynamic deployment involves iteratively moving a sensor based on virtual attractive or repulsive forces from other nodes, obstacles in the field and the areas that need higher coverage probability. Virtual force vectors depend on the distance between nodes and whatever attract or repulse them, and their relative directions. A sensor's new positions are computed in such a way that it moves in the direction of the virtual force by a step size proportional to its magnitude.

In [16], a $2n$-dimensional particle i represents x and y coordinates of all n mobile sensor nodes: $X_i = \{x_{i1}^1, x_{i1}^2, x_{i2}^1, x_{i2}^2, \cdots x_{in}^1, x_{in}^2\}$. The objective function $f(X_i)$ is the effective coverage, which the PSO maximizes. In order to achieve better coverage, the PSO velocity equation is modified to by adding the term $c_3 \cdot r_3(k) \cdot g_{ij}(k)$ to (1), where, c_3 is an acceleration constant, $r_3(k)$ is a random number uniformly distributed in [0,1], and g_{ij} is the set of new locations of n sensors computed using virtual forces method. *VFCPSO* combines advantages of virtual force and PSO. Here, the $2n$-dimensional PSO is converted into $2n$ single-dimensional PSOs, each conducted with an individual swarm. The final solution is produced by concatenating the $2n$ $gbest$ solutions. Authors report superior sensor coverage with significantly lesser computational effort. The method involves significant energy expenditure in broadcasting initial and final positions. It also necessitates mechanisms for localization and collision avoidance.

C. Base Station Positioning

Hong *et al*. have *PSO Multi-Base* for optimal positioning of multiple base stations in a two tier WSN [17]. The two tier network consists of nodes that can communicate only with the application nodes they are assigned to. Application nodes possess long-range transmitters, high-speed processors, and abundant energy. The *PSO Multi-Base* method aims at determining positions of base stations so that the total of distances of application nodes to their nearest base stations is minimum. This deployment requires minimum transmission power and, assures maximum network life. In *PSO Multi-Base*, a particle i represents the positions of M base stations, which can be in 2 or 3 dimensions based on the deployment terrain. The fitness of i is defined as $f(i) = \min(\sum_{j=1}^{n} l_{ij})$, where N is the number of application nodes. Here, l_{ij} represents the total lifetime of the network, as computed by $l_{ij} = \max \sum_{k=1}^{M} l_{i(k)j}$, the lifetime of the application node j that communicates with the base station k. The lifetime l_{ij} is computed as $l_{i(k)j} = \frac{e_j(0)}{r_j(\alpha_{j1}+\alpha_{j2} d_{i(k)j}^n)}$. Here $d_{i(k)j}^n$ represents the n^{th} order Euclidean distance from k^{th} base station to j^{th} application node. $e(0)$ is the initial energy, and α_1 and α_2 are the distance independent and distance dependent parameters that decide the energy necessary for the transmission respectively. While both *PSO Multi-Base* and exhaustive grid scan methods result in comparable lifetime, PSO converges in over 5 orders lesser time. The method is central, and needs location awareness. Besides, nodes have to communicate their initial energy to the base station; this energy overhead affects network scalability.

Summary: Static deployment is a one-time process in which solution quality is more important than fast convergence. PSO suits centralized deployment. Fast PSO variants are necessary dynamic deployment. PSO can also limit network scalability.

IV. Node Localization in WSNs

Node localization refers to creating location awareness in deployed nodes [18]. Location information is used in geometric-aware routing [19]. An obvious method of localization is to equip each node with a global positioning system (GPS), which is not attractive because of cost, size and power constraints. Many WSN localization algorithms estimate locations using *a priori* knowledge of the coordinates of special nodes called beacons, landmarks, or anchors. WSN localization is a two phase process. In ranging phase, nodes estimate their distances from beacons using signal propagation time or strength of the received signal. Signal propagation time is estimated through measurement of time of arrival, round trip time of flight or time difference of arrival of the signal [20]. Precise measurement of these parameters is not possible due to noise; therefore, results of such localization is inaccurate as shown in Figure 1. In the estimation phase, position of the target nodes is estimated using the ranging information either by solving simultaneous equations, or by an optimization algorithm that minimizes localization error. PSO algorithms for WSN localization are summarized in Table I.

A. Determination of Locations of Target Nodes

Gopakumar *et al*. have proposed *PSO-Loc* for localization of n target nodes out of m nodes based on the *a priori* information of locations of $m-n$ beacons [21]. The base station runs a $2n$-dimensional PSO (x and y coordinates of n nodes) to minimize the localization error defined as $f(x,y) = \frac{1}{M}\sum_{i=1}^{M}\left(\sqrt{(x-x_i)^2+(y-y_i)^2} - \hat{d}_i\right)^2$. Here, (x, y) is an estimate of the target node location, (x_i, y_i) is the location of beacon node i, and $M \geq 3$ is the number of beacons in the neighborhood of the target node. Estimated distance from beacon i, $\hat{d}_i$, is simulated as the actual distance corrupted by an additive Gaussian white noise. The variance of noise influences the localization accuracy. The approach does not take into account the issues of flip ambiguity and localization of the nodes that do not have at least three beacons in their neighborhood. The scheme works well only if either beacons have sufficient range, or there exist a large number of beacons. Moreover, the base station requires range estimates of all target nodes from all beacons in their neighborhoods. This requires

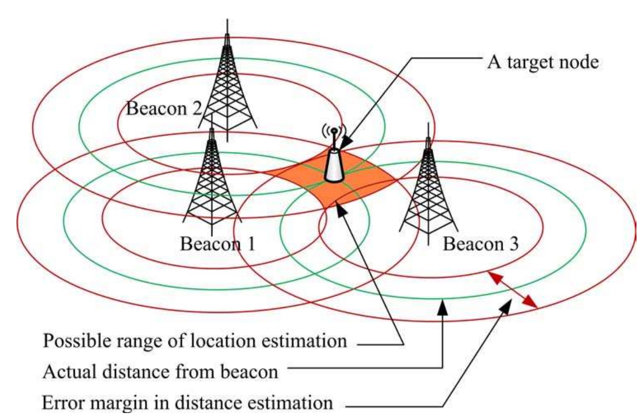

Fig. 1. Distance-based localization in a WSN

a lot of communication that may lead to congestions, delays and exhaustion of energy. In addition, the proposed scheme has a limited scalability because the PSO dimensionality is twice the number of target nodes.

PSO-Iterative: Kulkarni *et al*. have proposed a distributed iterative localization algorithm *PSO-Iterative* in [11]. Each target node that has three or more beacons in its hearing range runs PSO to minimize the localization error. Nodes that get localized act as beacons for other nodes. This continues iteratively, until either all the nodes get localized, or no more nodes can be. This method does not require that each node transmit it range measurement to a central node. Besides, it can localize all nodes that have three localized nodes or beacons in their range. As the localization iterations pass by, a node may get more number of references for localization, which mitigates the flip ambiguity problem, the situation that results in large localization error when the references are near-collinear. However, the proposed method is prone to error accumulation.

PSO-Beaconless: Low *et al*. have proposed in [22] a PSO-based distributed localization scheme that does not involve beacons. The nodes are deployed by an unmanned aerial vehicle equipped with a position sensor. The exact location Φ_i of a node i is treated as the conditional probability density function of Φ_{di}, the location where the node is deployed (which is recorded by the use of a pedometer). If this node can receive a signal from a localized node j it can estimate its distance d_j. A likelihood function for exact location is expressed in terms of Φ_{di} and d_j. PSO minimizes one term of this likelihood function. The results of two variants of the algorithm are presented. Results show fairly accurate localization even in sparse deployment. Authors report the results of real-time field tests of an implementation of the PSO-beaconless algorithm on a low-cost Microchip-PIC18LF4620 microcontroller [23]. It is reported that PSO takes longer computational time, but performs as accurate localization as the Gauss-Newton algorithm does when the pedometer accuracy is high. However, in less accurate pedometer records, the PSO outperforms the Gauss-Newton method in terms of localization accuracy.

PSO-4 Beacon: Low *et al*. have proposed *PSO-4 Beacon* localization scheme in [24]. This scheme assumes a presence of four beacons deployed roughly on boundaries of the sensor field. All target nodes can receive the signals from the beacons deployed at positions A, B, C and D. A node at location O in the sensor field can estimate its distance from a beacon as $d = (\frac{P}{P_0})^{-\frac{1}{\alpha}}$, where P is the power transmitted by the beacon and P_0 is the power at unit distance d_0. Environmental path loss exponent α plays an important role in distance estimation from the received signal strength. In the scheme proposed in [24], the target node at location O localizes by solving geometrical equations if the value of α is known. The target node uses PSO to find the best value of α and uses a Kalman filter based recursive estimation to localize itself. The paper reports fairly good localization accuracy.

Summary: Localization is a one-time optimization process in which solution quality is more important than fast convergence. Distributed localization is desirable due to energy issues. Though PSO is appropriate for distributed localization, the choice is influenced by availability of memory on the nodes.

V. Energy-Aware Clustering (EAC) in WSNs

Economic usage of energy is a critical issue in WSNs. Communication is the most energy expensive activity a node performs. Energy required to transmit varies exponentially with transmission distance; therefore, it is customary to use multi-hop communication in WSNs. A WSN's life-time largely depends on how efficiently it carries a data packet from its source to its destination. Routing refers to determining a path for a packet from a source node to a sink. The WSN that uses hierarchical routing has its nodes clustered into groups. Each cluster has a node that acts as the cluster-head. Nodes that belong to a cluster transmit their data packets to the cluster-head, which forwards it to the base station as shown in Figure 2. A node that acts as a cluster-head for a long duration exhausts its batteries prematurely. This calls for an optimal cluster-head election mechanism. Besides cluster assignment influences network performance and longevity. Low energy aware clustering hierarchy (LEACH) is a simple and efficient algorithm [25]. Clustering is an NP-hard optimization problem, which PSO can handle efficiently. Clustering or cluster-head selection is not a one-time activity; therefore, the simpler the optimization algorithm, the better the network efficiency is. This is another reason why PSO is a popular choice for WSN clustering. A summary of recent PSO applications in WSN clustering is given in Table I.

PSO-Clustering: Guru *et al*. have proposed four variants of PSO, namely, PSO with time varying inertia weight (PSO-TVIW), PSO with time varying acceleration constants (PSO-TVAC), hierarchical PSO with time varying acceleration constants (HPSO-TVAC) and PSO with supervisor student mode (PSO-SSM) for energy aware clustering in [26]. PSO assigns n_j nodes to each of the k cluster-heads, $j = 1, 2, \cdots, k$ such that the total energy loss due to physical distances E_{dd} is minimum. This is defined in (3), where D_j

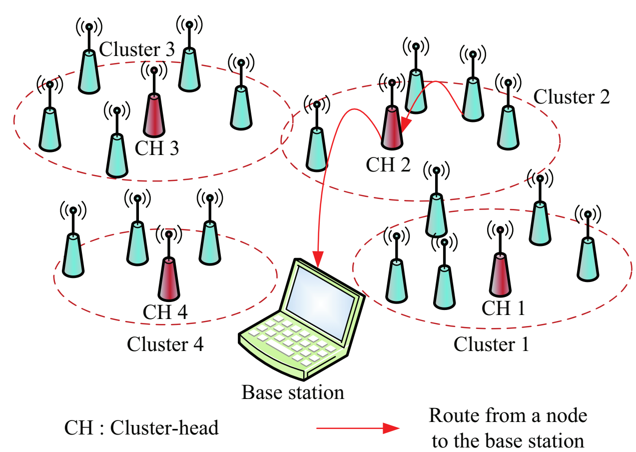

Fig. 2. The structure of a clustered WSN

is the distance between cluster-head j and the base station.

$$F = \sum_{j=1}^{k} \sum_{i=1}^{n_j} (d_{ij}^2 + \frac{D_j^2}{n_j}) \tag{3}$$

In PSO-TVIW, the inertia weight w is decreased linearly in each iteration. In PSO-TVAC, inertia weight is set constant, and acceleration constants c_1 and c_2 are varied linearly in every iteration. In HPSO-TVAC, the particle update is not influenced by the velocity in previous iteration; but, re-initialization of velocity is done when the velocity stagnates in the search space. Lastly, in PSO-SSM, the PSO update equation is modified to (4), where mc is a constant called momentum factor. Clustering is based on a simple idea that for a group of nodes that lie in a neighborhood, the node closest to the base station becomes the cluster-head. A detailed comparative analysis of the algorithms for optimal clustering is presented. This scheme considers only the physical distances between nodes and their assigned cluster-heads, but not the energy available to the nodes.

$$X_{id}(k+1) = (1 - mc) \cdot X_{id}(k) + mc \cdot V_{id}(k+1) \tag{4}$$

PSO-C: Latiff *et al*. consider both energy available to nodes and physical distances between the nodes and their cluster-heads in [27]. Each particle represents a combination of cluster-heads. The fitness function for the centralized PSO (PSO-C) is defined as $f = \beta \cdot f_1 + (1 - \beta) \cdot f_2$, where f_1 is the maximum average Euclidean distance of nodes to their associated cluster heads and f_2 is the ratio of total initial energy of all nodes to the total energy of the cluster-head candidates. These are expressed as (5) and (6) respectively.

$$f_1 = \max_{k=1,2,..K} \left\{ \sum_{\forall n_i \in C_{p,k}} \frac{d(n_i, CH_{p,k})}{|C_{p,k}|} \right\} \tag{5}$$

$$f_2 = \frac{\sum_{i=1}^{N} E(n_i)}{\sum_{k=1}^{K} E(CH_{p,k})} \tag{6}$$

Here, N is the number of nodes out of which K will be elected as the cluster-heads. $|C_{p,k}|$ is the number of nodes that belong to cluster C_k in particle p. This ensures that only the nodes that have above average energy resources are elected as the cluster-heads, and that the average distance between the nodes and the cluster-heads is minimum. They compare the results of the algorithm with those of LEACH and the LEACH-C algorithms [28]. The PSO-based clustering outperforms both LEACH and LEACH-C in terms of the network lifespan and the throughput. In [9], the authors show that this PSO-based algorithm outperforms GA and K-means-based clustering algorithms as well.

MST-PSO: Cao *et al*. have considered an interesting case in which a node and its cluster-head engage in a multi-hop communication [29]. The method computes a distance based minimum spanning tree of the weighted graph of the WSN. The best route between a node and its cluster-head is searched from all the optimal trees on the criterion of energy consumption. Cluster-heads are elected based on the energy available to the nodes and the Euclidean distance to its neighbor node in the optimal tree. The authors compare the performances of three mechanisms of cluster-head election: energy based, auto-rotation based and probability-based. Routing and cluster-head rotation are treated as optimization problems and tackled through PSO. The results show that the PSO-based clustering methods ensure longer network life.

Summary: Optimal clustering has a strong influence on the performance of WSN. Clustering is a centralized optimization carried out in a resource rich base station suitable for.

VI. Data-aggregation in WSNs

Large-scale deployment of sensors results in voluminous distributed data. Efficient collection of data is critical. Data-aggregation is the process of combining the data originating from multiple sources such that the result is better (more concise, more reliable etc) or the communication overhead is reduced [30]. A major application of a distributed WSN is to detect an event. In decentralized detection framework, each sensor node collects local observations corrupted by noise and sends a summary (compressed or partially processed data) to a fusion center. The fusion center uses the same to make the final global decision. This ensures an extended network lifespan at the expense of a reduction in performance. PSO has provided optimization in several aspects of data-aggregation as summarized in Table I.

A. Optimal Transmission Power Allocation

The wireless channel common to the nodes and the fusion center undergoes fading, which influences the accuracy of fusion. It is shown that the transmission power allocation scheme for distributed nodes plays an important role in the fusion error probability. Wimalajeewa *et al* address the problem of optimal power allocation through constrained PSO in [31]. Their algorithm *PSO-Opt-Alloc* uses PSO to determine optimal power allocation in the cases of both independent and correlated observations. The objective is to minimize the energy expenditure while keeping the fusion error probability under a required threshold. The authors present numerical results to show that the power schedule determined by PSO results in substantial energy savings in comparison to the uniform power schedule, especially in case of a large number of nodes.

B. Determination of Optimal Local Thresholds

In binary hypothesis-testing, distributed sensors make a binary (0 or 1) decision using local thresholds and send their decisions to a neighboring node. In a parallel fusion architecture, all nodes send their decisions to a base station; and in serial architecture, decisions follow a hop sequence from the first node to the base station. Thresholding leads to a gain in terms of bandwidth and energy, and a loss in terms of accuracy. Optimal thresholds on all nodes and an optimal

TABLE I
A SUMMARY OF APPLICATIONS OF PSO IN WSN DEPLOYMENT

	Task of PSO	Optimization criterion	Algorithm	Ref.	Centralized/ Distributed	Study
Deployment	Position stationary nodes	Max. coverage	PSO-Voronoi*	[13]	Centralized	Simulation
	Position stationary nodes	Min. cost of sensor equipment	PSO-Traffic*	[14]	Centralized	Real Deployment
	Position mobile nodes	Max. coverage	PSGO	[15]	Centralized	Simulation
	Position mobile nodes	Max. coverage	VFCPSO	[16]	Combination of both	Simulation
	Position base stations	Max. energy efficiency	PSO Multi-Base*	[17]	Centralized	Simulation
Localization	Localize nodes	Min. localization error	PSO-Loc*	[21]	Centralized	Simulation
	Localize nodes	Min. localization error	PSO-Iterative	[11]	Distributed	Simulation
	Localize nodes	Min. localization error	PSO-Beaconless*	[22]	Distributed	Simulation
	Localize nodes	Min. localization error	PSO-Beaconless*	[23]	Distributed	Real Deployment
	Localize nodes	Min. localization error	PSO-4 Beacon*	[24]	Distributed	Simulation
EAC	Elect cluster-heads	Min. intra-cluster distance	PSO-Clustering*	[26]	Centralized	Simulation
	Elect cluster-heads	Max. network longevity	PSO-C*	[27]	Centralized	Simulation
	Elect cluster-heads	Max. network longevity	MSTree-PSO*	[29]	Centralized	Simulation
Data-fusion	Allocate optimal transmission power	Min. energy expenditure and error probability	PSO-Opt-Alloc*	[31]	Centralized	Simulation
	Determine local thresholds	Min. decision error	ABC-PSO	[32]	Centralized	Simulation
	Determine sensor configuration	Min. decision error and transaction time	BMPSO	[33]	Distributed	Simulation
* Authors refer to this algorithm by this name						

decision route (called hierarchy) assure minimum energy expenditure and maximum accuracy. Veeramachaneni *et al*. present a hybrid of ant-based control and PSO (*ABC-PSO*) for hierarchy and threshold management [32]. In ant-based optimization, artificial ants move from a node to another constructing a partial solution to the problem. Once an ant reaches the final node, the performance of the solution is evaluated and the path emphasized using a mathematical value proportional to its performance (called pheromone). In *ABC-PSO* algorithm, ants construct the sequence and PSO identifies the thresholds and achieves the minimum error for the sequence. A feedback on this is presented to ants to help them move in the search space and identify better sequences.

C. Optimal Sensor Configuration

Multi-sensor systems consist of several sensing options and configurations. Adaptive configuration of the system having various sensor resources and multiple sensor parameters is a multi objective optimization problem. Objectives generally include maximum accuracy, minimum usage of communication resource, and maximum sensing coverage. Veeramachaneni *et al*. present a binary multi-objective PSO *BMPSO* for optimal configuration in [33]. This method uses Bayesian decision fusion framework to fuse the decisions from multiple sensors. Swarm agents are used to evolve the choice of sensors (each agent is a subset of sensors used for fusion). Each agent evokes PSO to evolve the thresholds and optimum fusion rules for its sensor set. The results highlight agents' ability to decide an optimal configuration of sensors, their thresholds and the optimal fusion rule.

Summary: Data-aggregation is a distributed repetitive process moderately suitable for PSO. Effective data-aggregation influences overall WSN performance and demands quick-convergence optimization techniques that assure high quality solutions. PSO is moderately suitable for this challenge.

VII. CONCLUSION

Scale and density of deployment, environmental uncertainties and constraints in energy, memory, bandwidth and computing resources pose serious challenges to the developers of WSNs. Issues of node deployment, localization, energy-aware clustering, and data-aggregation are often formulated as optimization problems. Most analytical methods suffer from slow or lack of convergence to the final solutions. This calls for fast optimization algorithms that produce quality solutions utilizing less resources. PSO has been a popular technique used to solve optimization problems in WSNs due to its simplicity, high quality of solution, fast convergence and insignificant computational burden. However, iterative nature of PSO can prohibit its use for high-speed real-time applications, especially if optimization needs to be carried out frequently. PSO requires large amounts of memory, which may limit its implementation to resource-rich base stations. Literature has abundant successful WSN applications that exploit advantages of PSO. Data-aggregation needs frequent distributed optimization, and fast solutions: Thus PSO moderately suits it. Static deployment, localization and clustering are the problems solved just once on a base station: Thus PSO highly suits them. Future research on PSO in WSN applications is likely to focus on:

1) Transformation of existing simulations into real-world applications.
2) Development of PSO in hardware.
3) Development of parameterless black-box PSO.
4) Cross-layer optimization through PSO.

An overview of PSO, issues in WSNs and a brief survey of recent PSO-based solutions to the WSN issues are presented in this paper. Advantages and limitations of PSO have been pointed out. A qualitative discussion on how PSO is tailored for WSN applications is presented, and promising research directions are projected. From the current rate of growth of PSO-based applications, it is envisioned that PSO will continue as an important optimization technique in several engineering fields including WSNs.

REFERENCES

[1] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292 – 2330, 2008.

[2] K. Romer and F. Mattern, "The design space of wireless sensor networks," *IEEE Trans. Wireless Commun.*, vol. 11, no. 6, pp. 54–61, 2004.

[3] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, vol. 4, 27 Nov.–1 Dec. 1995, pp. 1942–1948.

[4] Y. del Valle, G. K. Venayagamoorthy, S. Mohagheghi, J. C. Hernandez, and R. Harley, "Particle swarm optimization: Basic concepts, variants and applications in power systems," *IEEE Trans. Evol. Comput.*, vol. 12, no. 2, pp. 171–195, April 2008.

[5] R. Schaefer, *Foundations of Global Genetic Optimization*. Springer Publishing Company, Inc., 2007.

[6] K. V. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*, ser. Natural Computing Series. Berlin, Germany: Springer-Verlag, 2005.

[7] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," *IEEE Control Syst. Mag.*, vol. 22, no. 3, pp. 52–67, June 2002.

[8] R. Hassan, B. Cohanim, and O. de Weck, "A comparison of particle swarm optimization and the genetic algorithm," in *Proceedings of the AIAA/ASME/ASCE/AHS/ASC 46th Structures, Structural Dynamics and Materials Conference*, Austin, TX, Apr. 2005.

[9] N. M. A. Latiff, C. C. Tsimenidis, and B. S. Sharif, "Performance comparison of optimization algorithms for clustering in wireless sensor networks," in *Proceedings of the IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, 8–11 Oct. 2007, pp. 1–4.

[10] J. Vesterstrom and R. Thomsen, "A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems," in *Proceedings of Congress on Evolutionary Computation (CEC)*, vol. 2, June 2004.

[11] R. V. Kulkarni, G. K. Venayagamoorthy, and M. X. Cheng, "Bio-inspired node localization in wireless sensor networks," in *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, San Antonio, TX, Oct. 2009.

[12] Z. Bojkovic and B. Bakmaz, "A survey on wireless sensor networks deployment," *WSEAS Transactions on Communication*, vol. 7, no. 12, pp. 1172–1181, 2008.

[13] N. A. B. A. Aziz, A. W. Mohemmed, and B. S. D. Sagar, "Particle swarm optimization and Voronoi diagram for wireless sensor networks coverage optimization," in *Proceedings of the International Conference on Intelligent and Advanced Systems (ICIAS)*, 2007, pp. 961–965.

[14] J. Hu, J. Song, M. Zhang, and X. Kang, "Topology optimization for urban traffic sensor network," *Tsinghua Science & Technology*, vol. 13, Apr 2008.

[15] J. Li, K. Li, and W. Zhu, "Improving sensing coverage of wireless sensor networks by employing mobile robots," in *Proceedings of the International Conference on Robotics and Biomimetics (ROBIO)*, 2007, pp. 899–903.

[16] X. Wang, S. Wang, and J. J. Ma, "An improved co-evolutionary particle swarm optimization for wireless sensor networks with dynamic deployment," *Sensors*, vol. 7, pp. 354–370, 2007.

[17] T. P. Hong and G. N. Shiu, "Allocating multiple base stations under general power consumption by the particle swarm optimization," in *Proceedings of the IEEE Swarm Intelligence Symposium (SIS)*, 2007, pp. 23–28.

[18] A. Boukerche, H. A. B. Oliveira, E. F. Nakamura, and A. A. F. Loureiro, "Localization systems for wireless sensor networks," *IEEE Wireless Commun. Mag.*, vol. 14, no. 6, pp. 6–12, December 2007.

[19] J. Aspnes, T. Eren, D. K. Goldenberg, A. S. Morse, W. Whiteley, Y. R. Yang, B. D. O. Anderson, and P. N. Belhumeur, "A theory of network localization," *IEEE Trans. Mobile Comput.*, vol. 5, no. 12, pp. 1663–1678, Dec. 2006.

[20] G. Mao, B. Fidan, and B. D. O. Anderson, "Wireless sensor network localization techniques," *Computer Networks*, vol. 51, no. 10, pp. 2529–2553, 2007.

[21] A. Gopakumar and L. Jacob, "Localization in wireless sensor networks using particle swarm optimization," in *Proceedings of the IET International Conference on Wireless, Mobile and Multimedia Networks*, 2008, pp. 227–230.

[22] K. S. Low, H. A. Nguyen, and H. Guo, "A particle swarm optimization approach for the localization of a wireless sensor network," in *Proceedings of the IEEE International Symposium on Industrial Electronics (ISIE)*, 2008, pp. 1820–1825.

[23] H. Guo, K. S. Low, and H. A. Nguyen, "Optimizing the localization of a wireless sensor network in real time based on a low cost microcontroller," *IEEE Trans. Ind. Electron.*, pp. 1–9, In Press.

[24] K. S. Low, H. A. Nguyen, and H. Guo, "Optimization of sensor node locations in a wireless sensor network," in *Proceedings of the 4th International Conference on Natural Computation (ICNC)*, vol. 5, 2008, pp. 286–290.

[25] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences*, vol. 2, 2000.

[26] S. Guru, S. Halgamuge, and S. Fernando, "Particle swarm optimisers for cluster formation in wireless sensor networks," in *Proceedings of the International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP)*, S. K. Halgamuge, Ed., 2005, pp. 319–324.

[27] N. M. A. Latiff, C. C. Tsimenidis, and B. S. Sharif, "Energy-aware clustering for wireless sensor networks using particle swarm optimization," in *Proceedings of the 18th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, 2007, pp. 1–5.

[28] P. Chandrakasan and W. B. Heinzelman, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. Wireless Commun.*, vol. 1, no. 4, pp. 660–670, Oct. 2002.

[29] X. Cao, H. Zhang, J. Shi, and G. Cui, "Cluster heads election analysis for multi-hop wireless sensor networks based on weighted graph and particle swarm optimization," in *Proceedings of the 4th International Conference on Natural Computation (ICNC)*, vol. 7, 2008, pp. 599–603.

[30] R. Rajagopalan and P. K. Varshney, "Data-aggregation techniques in sensor networks: A survey," *IEEE Commun. Surveys Tuts.*, vol. 8, no. 4, pp. 48–63, Fourth Quarter 2006.

[31] T. Wimalajeewa and S. K. Jayaweera, "Optimal power scheduling for correlated data fusion in wireless sensor networks via constrained PSO," *IEEE Trans. Wireless Commun.*, vol. 7, no. 9, pp. 3608–3618, 2008.

[32] K. Veeramachaneni and L. A. Osadciw, "Swarm intelligence based optimization and control of decentralized serial sensor networks," in *Proceedings of the IEEE Swarm Intelligence Symposium*, 2008, pp. 1–8.

[33] K. Veeramachaneni and L. A. Osadciw, "Dynamic sensor management using multi objective particle swarm optimizer," in *Swarm Optimizer, SPIE Defence and Security Symposium, April 16-20*, 2004.

Raghavendra V. Kulkarni (M'97–SM'05) received his PhD degree in electrical engineering from the Missouri University of Science and Technology (Missouri S&T), Rolla, USA, in April 2010. His research interests include the development of wireless sensor network applications using computational intelligence tools. He is a life member of the Indian Society for Technical Education (ISTE), and a member of the IEEE Computational Intelligence Society and the International Neural Network Society (INNS).

Ganesh Kumar Venayagamoorthy (S'91–M'97–SM'02) received his PhD degree in electrical engineering from the University of KwaZulu Natal, Durban, South Africa, in Feb. 2002. Currently, he is an Associate Professor of Electrical and Computer Engineering, and the Founder and Director of the Real-Time Power and Intelligent Systems (RTPIS) Laboratory at Missouri S&T. His research interests are in the development of advanced computational algorithms for real-world applications, including power systems stability and control, smart grid applications, sensor networks and signal processing. He has published 2 edited books, 6 book chapters, over 85 refereed journals papers and 275 refereed conference proceeding papers. He has been involved in approximately US$ 7 Million of competitive research funding.