01 Sep 2008

Network-centric Localization in MANETs Based on Particle Swarm Optimization

Raghavendra V. Kulkarni

Ganesh K. Venayagamoorthy
Missouri University of Science and Technology

Ann K. Miller
Missouri University of Science and Technology

Cihan H. Dagli
Missouri University of Science and Technology, dagli@mst.edu

Follow this and additional works at: https://scholarsmine.mst.edu/engman_syseng_facwork

Part of the Electrical and Computer Engineering Commons, and the Systems Engineering Commons

Recommended Citation

Network-centric Localization in MANETs based on

Particle Swarm Optimization

Raghavendra V. Kulkarni, *Senior Member, IEEE,* Ganesh K. Venayagamoorthy, *Senior Member, IEEE,*
Ann Miller, *Senior Member, IEEE,* and Cihan H. Dagli, *Member, IEEE*

Abstract—There exist several application scenarios of mobile ad hoc networks (MANET) in which the nodes need to locate a target or surround it. Severe resource constraints in MANETs call for energy efficient target localization and collaborative navigation. Centralized control of MANET nodes is not an attractive solution due to its high network utilization that can result in congestions and delays. In nature, many colonies of biological species (such as a flock of birds) can achieve effective collaborative navigation without any centralized control. Particle swarm optimization (PSO), a popular swarm intelligence approach that models social dynamics of a biological swarm is proposed in this paper for network-centric target localization in MANETs that are enhanced by mobile robots. Simulation study of two application scenarios is conducted. While one scenario focuses on quick target localization, the other aims at convergence of MANET nodes around the target. Reduction of swarm size during PSO search is proposed for accelerated convergence. The results of the study show that the proposed algorithm is effective in network-centric collaborative navigation. Emergence of converging behavior of MANET nodes is observed.

Index Terms—Mobile ad hoc networks (MANETs), multi-robot systems, network-centric control, particle swarm optimization (PSO), target localization, wireless sensor networks (WSNs)

I. Introduction

WIRELESS sensor network (WSN) is a network of distributed autonomous devices that can sense or monitor physical or environmental conditions cooperatively [1]. Sensor nodes that form WSNs are deployed in an ad hoc manner for remote operations. Applications of WSNs include environmental monitoring, habitat monitoring, prediction and detection of natural calamities, medical monitoring and structural health monitoring [2]. WSNs are expected to realize a convergence of communication, computing, and control. Sensor nodes are typically small and inexpensive, operating with limited resources, often in adverse stochastic environments. Sensor nodes have stringent limitations in storage resources, computational capabilities, communication bandwidth and power

supply. These constraints call for a distributed control. In some applications, sensor nodes are grouped in clusters, and each cluster has a node that acts as the cluster head. All nodes in a cluster forward their sensor data to the cluster head, which in turn routes it to a specialized node called the sink node (or the base station) through a multi-hop wireless communication. Self organization, energy efficient routing, scheduling, security and reliability are the key research topics in the area of WSNs [2]- [3].

Nodes of a WSN are often endowed with mobility in order to expand the spectrum of its capabilities. A mobile ad hoc network (MANET) is a network of mobile devices that can communicate with each other without the use of a predefined infrastructure. A popular application of MANETs is to provide service as the first respondents in emergency situations [4]. WSNs can be regarded as a subset of MANETs due to the fact that the nodes are stationary in most sensor networks. Generally, MANETs consist of WSNs that are enhanced by mobile platforms. In most applications, these mobile platforms are mobile robot systems [5]. Fragile, low bandwidth wireless links and frequent topology changes pose challenges in network discovery, network control and routing, collaborative information processing, querying, and tasking [2]. Operation and control of nodes in a WSN or a MANET is a major research issue. A tutorial-style overview of cooperative control in sensor networks having stationary and mobile nodes can be found in [6]. Typically, central control is exercised for sensing, transmission, and locomotion, which may result in increased network load, leading to congestions and delays. These problems become more severe when the number of nodes increases. Though approaches like directed diffusion exist for data centric communication [7], there exist many unresolved research issues.

There are several application scenarios in which the nodes of a MANET need to be navigated to a desired location in the mission space for a closer investigation of the environment. Searching for a fire is one such example. The quicker a node locates the source of the fire, the quicker can the fire be extinguished. Navigation of a group of nodes (or all the nodes) to a desired target location is another requirement. For example, when a node locates the fire, it might need the assistance of many of its peers in extinguishing the fire. Many colonies of biological species such as swarms of bees and flocks of birds have mastered the art of collaborative navigation and convergence without using any centralized command. This paper discusses an approach for network centric operation and control for node localization in MANETs. The approach incorporates the elements of particle

swarm optimization (PSO), the collaborative problem solving technique that colonies of biological species use in nature. This approach generates in-network navigational decisions and obviates centralized control. This network centric approach is effective in a typical MANET scenario which involves node mobility, large deployment and energy constraints.

Main contributions of this paper are as follows: Two PSO based localization algorithms are presented. The first algorithm navigates the sensor nodes of a MANET to search for a target such as a fire or a source of odor. When the first node reaches a point in sufficiently close vicinity of the target, the mission is deemed accomplished. The second algorithm aims at quick convergence of a sizeable number of nodes around the target. Results of simulation studies are presented.

The rest of the paper is organized as follows: The potential of the network centric operations in MANETs and WSNs is discussed in section II. Section III reviews the approaches used for multi robot target localization. Fundamental concepts of PSO are explained in section IV. Details of the simulation studies are given and the results obtained are discussed in section V. Finally, conclusions are presented in section VI.

II. NETWORK-CENTRIC OPERATION AND CONTROL

Coordination and control of MANETs, especially collaborative target localization is an emerging research area. The MANETs monitor an environment or a phenomenon continuously. Sensor data is transmitted to a central system for processing through multi-hop communication. Then the navigation commands are delivered. This scheme is illustrated in Fig. 1. If data needs to be transmitted over a long distance, one has to either use a high power transmission, or several number of hops. While the former scheme results in premature exhaustion of the nodes, the latter results in accumulation of delays. Centralized control is prone to network congestion as well.

A possible way to overcome these limitations is to use an efficient network-centric localization scheme. The desired network-centric control and operations in a MANET is illustrated in Fig. 2. This scheme employs data-centric message forwarding, aggregation and processing. Major requirements of such a scheme are:
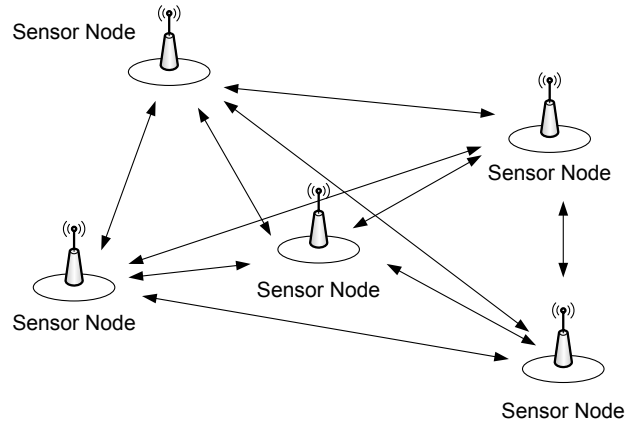

Fig. 1. Centralized control in a MANET

Fig. 2. Network centric operation and control in a MANET

1) Self organized operation without centralized control
2) Self learning properties
3) Reduced network utilization
4) Faster response

Scientists have sought inspiration from nature to address many technological challenges. Computer technology is full of mechanisms that have been adapted from biological systems. Bio-inspired technologies have been proposed for network-centric actuation in sensor/actuator networks [8]. Swarm intelligence (SI), a paradigm of computational intelligence, encompasses several bio-inspired techniques. PSO, a popular SI algorithm [9], is used for network-centric collaborative target localization in this study.

III. MULTI-ROBOT TARGET LOCALIZATION

It has long been recognized that there are several tasks that can be performed more efficiently and robustly using multiple robots. Moving from one robot to a group of robots increases the resources available to accomplish a task, but adds its own complexity. Locating one or more targets within an unknown environment is a task well-suited to a group of mobile robots. One major challenge within multi-robot target localization is to design effective algorithms that allow a team of robots to work together to find target locations. Literature has a large number of techniques proposed for efficient multi-robot navigation. An overview of cooperative multi-vehicle test-bed (COMET) created in order to facilitate the development of cooperative control systems and mobile sensor networks is presented in [10]. Video camera assisted vision based control [11], neuro-fuzzy control [12] and behavior-based control [13] of multiple robot platforms in real-time are some techniques reported in literature.

Use of swarm-intelligent robotic approach in search tasks can offer multiple benefits like parallel search, quick convergence and increased robustness against failure of a robot. SI techniques, the techniques based on the collective behavior of decentralized, self-organized systems, have been used to solve target localization problem. A swarm-based fuzzy logic control for collaboratively locating the hazardous contaminants in an unknown large-scale area is proposed in [14].
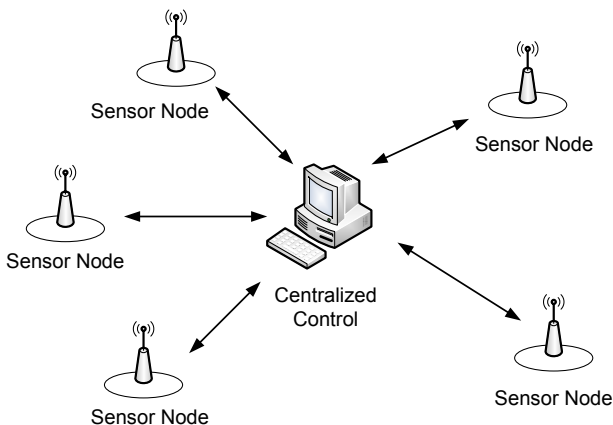
PSO uses a virtual multi-agent search to find global optima in a multi-dimensional solution space. It has been adapted for multi-robot target localization problem. PSO based multi-robot target search approach is presented in [15] and [16]. The work discussed in [17] proposes a PSO based mobile sensor network for odor source localization in an environment having a dynamic odor distribution. The work presented in this paper uses PSO to navigate the robots laden with sensor nodes of a MANET to a target location.

IV. Basic Concepts of PSO

PSO is a population based iterative parallel search algorithm that models social behavior of birds within a flock. PSO uses a simple concept, and it can be implemented in a few lines of computer code. It requires only primitive mathematical operators, and is inexpensive in terms of memory requirements and computational time. It has been found effective in solving several kinds of problems. Since its introduction in [9], PSO has seen many modifications and has been adapted to different complex environments [18]. Many versions of PSO have been proposed [19] and applied to solve optimization problems in such diverse fields as reactive power systems [19]- [20], stock markets [21], distribution state estimation [22] and adaptive phased array antenna control [23].

PSO consists of a population (or a swarm) of s particles, each of which represents a potential solution. The particles explore an n dimensional solution space in search of the global solution, where n represents the number of parameters to be optimized. Each particle i occupies a position X_{id} and moves with a velocity v_{id}, $1 \leq i \leq s$ and $1 \leq d \leq n$. The particles are initially assigned random positions and velocities within fixed boundaries, i.e., $X_{min} \leq X_{id} \leq X_{max}$ and $v_{min} \leq v_{id} \leq v_{max}$. Fitness of a particle is determined from its position. The fitness is defined in such a way that a particle closer to the global solution has higher fitness value than a particle that is far away. In each iteration, velocities and positions of all particles are updated to persuade them to achieve better fitness. The process of updating is repeated iteratively either until a particle reaches the global solution within permissible tolerance limits, or until a sufficiently large number of iterations is reached. Magnitude and direction of movement of a particle are influenced by its previous velocity, its experience and the knowledge it acquires from the swarm through social interaction.

Each particle has a memory where it stores $pbest_{id}$, the position where it had the highest fitness. Besides, $gbest_d$, the maximum of $pbest_{id}$s of all particles is stored as well. The $gbest$ particle represents the best solution found as yet. At each iteration k, PSO adds velocity v_{id} to the position X_{id} of a particle and steers the particle towards its $pbest_{id}$ and $gbest_d$ using (1) and (2).

$$v_{id}(k+1) = w \cdot v_{id}(k) + c_1 \cdot rand_1 \cdot (pbest_{id} - X_{id}) \\ + c_2 \cdot rand_2 \cdot (gbest_d - X_{id}) \quad (1)$$

$$X_{id}(k+1) = X_{id}(k) + v_{id}(k+1) \quad (2)$$

Here, $rand_1$ and $rand_2$ are the random numbers that range between 0 and 1 with a uniform distribution. The velocity update equation (1) shows that a particle is influenced by 3 components of acceleration. The first term involves the inertia coefficient w, $0.2 < w < 1.2$, and it denotes the inertia of the particle [24]. The second term involves the cognitive acceleration constant c_1. This component propels the particle towards the position where it had the highest fitness. The third term involves the social acceleration constant c_2. This component steers the particle towards the particle that currently has the highest fitness. Fig. 3 illustrates these components of velocity. The net change in a particle's velocity vector is equal to the vector sum of these individual velocities.

The velocity of a particle is bounded between properly chosen limits v_{max} and v_{min}. If v_{max} is a very large constant, particles might acquire a velocity high enough to go out of the solution space. If v_{max} is a very small constant, particles might move in very small steps and therefore, take a long time to converge to the solution. Similarly, the position of a particle is restricted between properly chosen constants X_{max} and X_{min}. Pseudocode for the PSO based search algorithm is given in Fig. 4.

V. Simulation Results and Discussion

In this study, two PSO based application scenarios are simulated in MATLAB software environment. The PSO particles are the MANET nodes enhanced by mobile robots. As done in some real-time MANETs, the nodes are randomly deployed in the mission space. The mission space is a 2-dimensional plane grid having an area of 1000×1000 square units. It is assumed that the nodes always know their current locations. The user defines a target by specifying its cartesian coordinates (T_x, T_y). However, the coordinates of the target are unknown to the sensors. It is assumed that each node has a sensor which provides with the information necessary to compute its distance from the target. The PSO seeks to minimize the distance to target (or to maximize the strength of sensor signal). The sensor is simulated in such a way that the detected signal strength E is obtained as the reciprocal of the sensor's
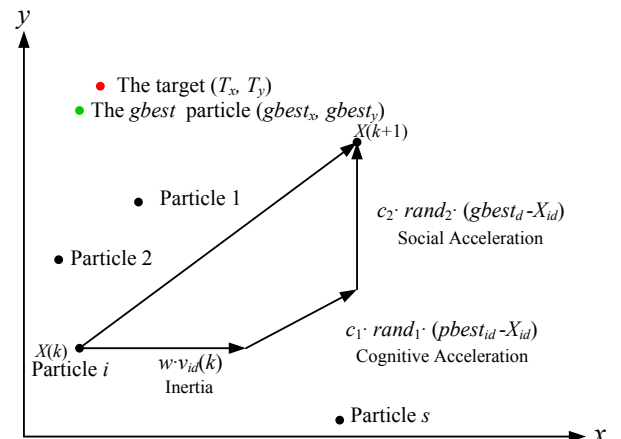

Fig. 3. Components of velocity update in PSO

```
Initialize w, c₁, c₂, max_iterations, target_fitness, Xmin, Xmax, vmin and vmax
FOR each particle i
    FOR each dimension d
        Initialize positions randomly, Xmin ≤ Xid ≤ Xmax
        Initialize velocity vid randomly, vmin ≤ vid ≤ vmax
    End FOR
END FOR
Iteration k = 0
DO
    FOR each particle i
        Compute fitness(i)
        IF fitness(i) > fitness(pbestd)
            FOR each dimension d
                pbestid = Xid
            End FOR
        END IF
        IF fitness(i) > fitness(gbestd)
            FOR each dimension d
                gbestd = Xid
            End FOR
        END IF
    END FOR

    FOR each particle i
        FOR each dimension d
            Calculate velocity according to the equation
            vid(k+1)= w·vid(k) + c₁·rand₁·(pbestid-Xid) + c₂·rand₂· (gbestd-Xid)
            Update particle position according to the equation
            Xid(k+1)= Xid(k) + vid(k+1)
            Restrict vid within vmin and vmax
            Restrict Xid within Xmin and Xmax
        END FOR
    END FOR
    k=k+1
WHILE k ≤ max_iterations AND fitness(gbest) < target_fitness
```
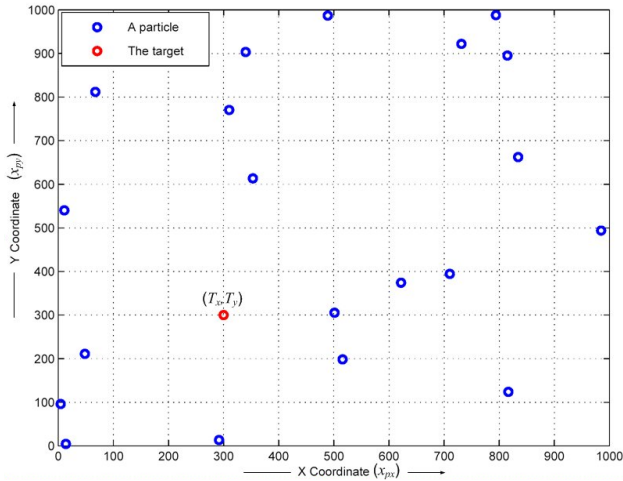
Fig. 4. Pseudocode for PSO

Fig. 5. The top view of the initial mission space

distance from the target. Strength of the signal detected by sensor i is obtained from (3).

$$E(i) = \frac{1}{\sqrt{(T_x - X_{ix})^2 + (T_y - X_{iy})^2}} \tag{3}$$

The top view of the mission space before the PSO starts is shown in Fig. 5.

The PSO simulated in this study has a population of 20 particles. Limits of particle position are X_{max}=1000, and X_{min}=0. Due to the planar mission space, the PSO particles have 2 dimensions. Therefore, the particle i is represented as a pair (X_{ix}, X_{iy}), where X_{ix} and X_{iy} represent horizontal (x)

and vertical (y) coordinates respectively. The particle velocity is bounded by v_{max}=20 and v_{min}=-20. The inertia weight w is chosen as 0.8; and the acceleration constants are chosen as $c_1 = c_2 = 2$. The results obtained in each of the scenarios are presented in the following subsections.

A. Scenario 1

In this scenario, the goal of MANET nodes is to locate the target. In real world, this task is identical to searching for a fire in the mission space. The search is terminated when a node finds the target location within the predefined tolerance limit. Here, the tolerance limit is taken as 0.01 units. The fitness, which the PSO seeks to maximize is the signal strength E, which is obtained from (3).

$$fitness(i) = E(i) \tag{4}$$

The average distance d_{av} of the swarm to the target is computed using (5).

$$d_{av} = \sum_{i=1}^{s} \sqrt{(T_x - X_{ix})^2 + (T_y - X_{iy})^2} \tag{5}$$

As the PSO search progresses, the particles move closer to the target. Fig. 6 shows the reduction in the distance between the target and the particles as the iterations progress.

25 trial runs are executed for the sake of statistical analysis of the results. Mean and standard deviations of the number for iterations taken for convergence, average swarm distance and fitness of the best particle before termination are recorded. Following is the statistical summary of the results obtained:

- Mean number of iterations = 105.25
- Standard deviation in the number of iterations = 18.89
- Mean of average swarm distance to the target = 53.73
- Standard deviation in average swarm distance to the target = 38.32
- Mean distance to the target from the best particle = 0.0064
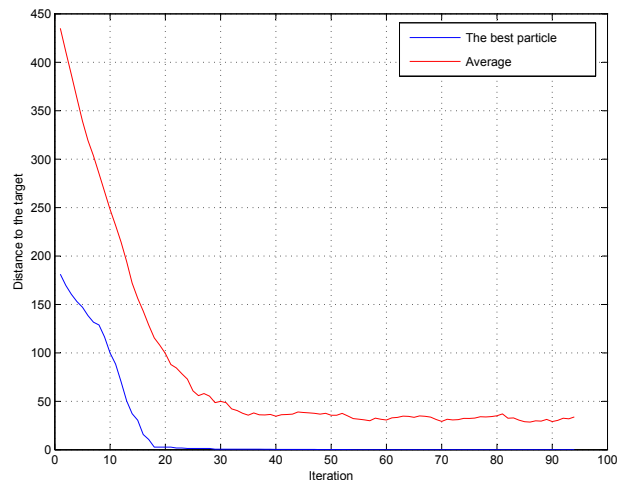- Standard deviation in distance to the target from the best particle = 0.0025

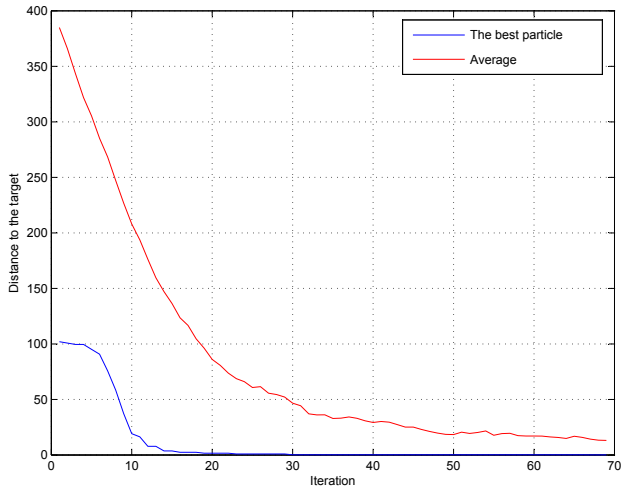Fig. 6. Iterative reduction of distance to the target in the scenario 1

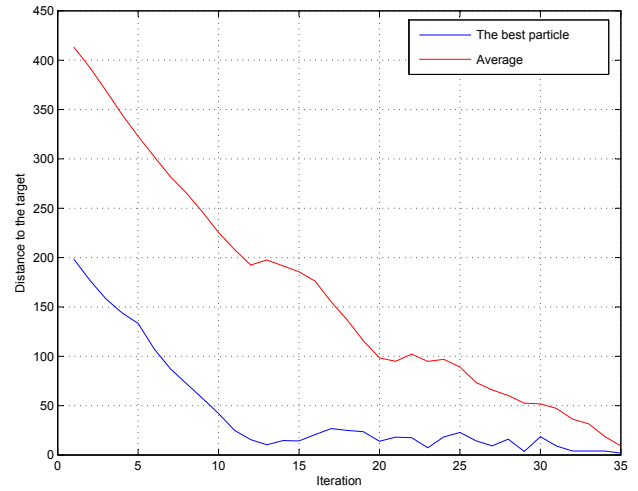Fig. 7. Iterative reduction of distance to the target in scenario 2, strategy 1

Fig. 8. Iterative reduction of distance to the target in scenario 2, strategy 2

B. Scenario 2

In this scenario, the goal of robots that carry the MANET nodes is to converge around the target location. In real applications, this task is similar to convergence of a group of fire-fighters around the fire to extinguish it quickly in a collaborative manner. Two strategies are proposed here to achieve the task and each strategy uses a different variant of PSO. In strategy 1, the PSO based navigation task is terminated when 75% of the nodes (15 nodes in this study) converge within a circle of 5 unit radius around the target. Fitness of a particle and the average distance of the swarm to the target are computed using (4) and (5) respectively. 25 trial runs are executed for the sake of statistical analysis of the results. Mean and standard deviations of the number for iterations taken for convergence and the average swarm distance are recorded. The statistical summary of the results obtained is given in Table I.

Fig. 8 shows the reduction in the distance between the target and the particles as the iterations progress.

The objective here is not to navigate a particle to go close to the target, but to navigate a group of particles to converge around the target. How close to the target the best particle goes has no relevance in this scenario. Therefore, the conventional PSO, which persuades the particles to move closer to the target, is not the perfectly suitable approach. In strategy 2, a modification is done to the PSO algorithm such that quicker convergence is achieved. When the $gbest$ particle reaches within the circle of 5 unit radius around the target location, it is excluded from the swarm; and the PSO navigation is continued with one less particle. This is repeated until 75% of the particles reach within the circle. Fig. 8 shows the reduction in the distance between the target and the particles as the iterations progress in the modified PSO. This modification enhances the convergence speed as indicated by the results of 25 trial runs which are summarized in Table I.

The particle that has the maximum fitness represents the best solution found by the PSO algorithm. The global best fitness of the swarm never decreases from an iteration to another.

This is clear from the plots of the distance to target from the best particles (reciprocal of the best fitness) in Figs. 6 and 7. However, several rises can be observed in the plot of distance to the target from the best particle in Fig. 8. This is obvious because, when the current best particle moves within the tolerance circle around the target, the particle from those outside, which is nearest to the periphery of the circle will be chosen as the next best particle.

The effectiveness of PSO in determining the solution to a multi-dimensional optimization problem is reflected by its ability to locate the target with an accuracy of 0.01 unit distance in an average of 105 iterations in scenario 1 of this study. The only external data the particles need to use is the coordinates and the fitness of the $gbest$ particle. Only the particle that attains better fitness than the current $gbest$ particle has to transmit this information, based on which the new $gbest$ particle will be elected. The total communication effort needed here is much less than in the centralized control scenario in which each particle would receive navigational control data from the central controller.

The modifications made to PSO in strategy 2 of scenario 2 resulted in faster convergence. The standard PSO used in strategy 1 achieved the convergence in 134.32 iterations, while the modified PSO achieved the same in 50.6 iterations. This saving comes at a cost of s comparisons in every iteration

TABLE I
SUMMARY OF RESULTS OF PSO AND MODIFIED PSO FOR SCENARIO 2

	Strategy 1: PSO	Strategy 2: Modified PSO	Difference
Mean number of iterations	134.32 (93.64)	50.60 (11.67)	62.32%
Mean of average swarm distance to the target (d_{av})	67.39 (54.32)	45.64 (28.61)	32.27%
Convergence Time	0.2462 (0.16)	0.1163 (0.16)	52.76%

The numbers shown in brackets represent standard deviation.
All trial runs are conducted on the same computer.

where fitness of each particle is compared with a constant before deciding if the particle should remain in the swarm. The plots of average distance from the swarm to the target show the emergence of a convergence behavior pattern.

VI. CONCLUSIONS

This paper presents an algorithm for network-centric target localization of the nodes of a MANET enhanced by mobile robots. The proposed navigation algorithm uses PSO, a popular bio-inspired, population based optimization technique. Two application scenarios are simulated in MATLAB software environment. While the first scenario aims at locating the target, the second scenario aims at convergence of a major section of population in the vicinity of the target. A modified PSO algorithm is proposed for quicker convergence in the second scenario. The performance analysis is done by results obtained in 25 trial runs of each scenario. It is demonstrated that the PSO enables collaborative navigation based on local intelligence. The interaction and collaboration between the MANET nodes results in an optimized swarm behavior in an emergent fashion.

Extension of this study is possible in several directions. The practical usability of the methods studied in the simulations needs to be assessed in real-time MANET applications. The simulation done here assumes almost ideal planar mission space without any obstacles. The collaborative navigation techniques that are suited for collision avoidance can be combined with the proposed algorithms. Besides, this work assumes that the MANET nodes know their locations, which may not be possible in some applications. Investigation of the effect of localization error is another direction in which the work can be extended. A detailed investigation of computational complexities of the methods is yet another area where the extension of this work is possible.

REFERENCES

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Commun. Mag.*, vol. 40, no. 8, pp. 102–114, Aug. 2002.

[2] C. Y. Chong and S. Kumar, "Sensor networks: Evolution, opportunities, and challenges," *Proc. IEEE*, vol. 91, no. 8, pp. 1247–1256, Aug. 2003.

[3] D. Culler, D. Estrin, and M. Srivastava, "Guest editors' introduction: Overview of sensor networks," *Computer*, vol. 37, no. 8, pp. 41–49, Aug. 2004.

[4] V. Kumar, D. Rus, and S. Singh, "Robot and sensor networks for first responders," *IEEE Pervasive Comput.*, vol. 3, no. 4, pp. 24–33, Oct–Dec 2004.

[5] L. E. Parker, "Current research in multirobot systems," *Journal of Artificial Life Robotics*, vol. 7, 2003.

[6] C. Cassandras and W. Li, "Sensor networks and cooperative control," in *Proc. and 2005 European Control Conference Decision and Control CDC-ECC '05. 44th IEEE Conference on*, 12–15 Dec. 2005, pp. 4237–4238.

[7] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks," in *MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM Press, 2000, pp. 56–67. [Online]. Available: http://portal.acm.org/citation.cfm?id=345920

[8] F. Dressler, "Network-centric actuation control in sensor/actuator networks based on bio-inspired technologies," in *Proc. IEEE International Conference on Mobile Ad hoc and Sensor Systems (MASS)*, Oct. 2006, pp. 680–684.

[9] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Networks*, vol. IV, Perth, Australia, Jan. 1995, pp. 1942–1948.

[10] D. Cruz, J. McClintock, B. Perteet, O. Orqueda, Y. Cao, and R. Fierro, "Decentralized cooperative control - A multivehicle platform for research in networked embedded systems," *IEEE Control Syst. Mag.*, vol. 27, no. 3, pp. 58–78, June 2007.

[11] M. Fiala, "Vision guided control of multiple robots," in *Proc. First Canadian Conference on Computer and Robot Vision*, May 17–19, 2004, pp. 241–246.

[12] K. Ng and M. Trivedi, "A neuro-fuzzy controller for mobile robot navigation and multirobot convoying," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 28, no. 6, pp. 829–840, 1998.

[13] S. Berman, Y. Edan, and M. Jamshidi, "Navigation of decentralized autonomous automatic guided vehicles in material handling," *IEEE Trans. Robot. Autom.*, vol. 19, no. 4, pp. 743–749, 2003.

[14] X. Cui, T. Hardin, R. Ragade, and A. Elmaghraby, "A swarm-based fuzzy logic control mobile sensor network for hazardous contaminants localization," in *Proc. IEEE International Conference on Mobile Ad-hoc and Sensor Systems*, 25–27 Oct. 2004, pp. 194–203.

[15] J. Pugh and A. Martinoli, "Inspiring and modeling multi-robot search with particle swarm optimization," in *Proc. IEEE Swarm Intelligence Symposium SIS 2007*, A. Martinoli, Ed., 2007, pp. 332–339.

[16] G. Venayagamoorthy and S. Doctor, "Navigation of mobile sensors using PSO and embedded PSO in a fuzzy logic controller," in *39th IAS Annual Meeting Industry Applications Conference Conference Record of the 2004 IEEE*, S. Doctor, Ed., vol. 2, 2004, pp. 1200–1206 vol.2.

[17] W. Jatmiko, K. Sekiyama, and T. Fukuda, "A PSO-based mobile sensor network for odor source localization in dynamic environment: Theory, simulation and measurement," in *Proc. CEC 2006. Evolutionary Computation IEEE Congress on*, K. Sekiyama, Ed., 2006, pp. 1036–1043.

[18] X. Hu, Y. Shi, and R. Eberhart, "Recent advances in particle swarm," in *Proc. CEC2004 Evolutionary Computation Congress on*, vol. 1, 19–23 June 2004, pp. 90–97.

[19] Y. del Valle, G. K. Venayagamoorthy, S. Mohagheghi, J. C. Hernandez, and R. G. 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.

[20] H. Yoshida, K. Kawata, and Y. Fukuyama, "A particle swarm optimization for reactive power and voltage control considering voltage security assessment," *IEEE Trans. Power Syst.*, vol. 15, pp. 1232–1239, Nov. 2000.

[21] J. Nenortaite and R. Simutis, "Adapting particle swarm optimization to stock markets," in *Proc. 5th International Conference on Intelligent Systems Design and Applications ISDA '05*, R. Simutis, Ed., 2005, pp. 520–525.

[22] S. Naka, T. Genji, T. Yura, and Y. Fukuyama, "Practical distribution state estimation using hybrid particle swarm optimization," in *Proc. IEEE Power Eng. Soc. Winter Meeting*, vol. 2, Columbus, OH, Jan. 2001, pp. 815–820.

[23] M. Donelli, R. Azaro, F. D. Natale, and A. Massa, "An innovative computational approach based on a particle swarm strategy for adaptive phased-arrays control," *IEEE Trans. on Antennas and Propagation*, vol. 54, no. 3, pp. 888–898, Mar. 2006.

[24] Y. Shi and R. C. Eberhart, "Parameter selection in particle swarm optimization," in *EP '98: Proceedings of the 7th International Conference on Evolutionary Programming VII*. London, UK: Springer-Verlag, 1998, pp. 591–600.