Cooperative Control Of A Robot Swarm With Network Communication Delay

Yechiel Crispin

Department of Aerospace Engineering,
Embry-Riddle University, Daytona Beach, FL 32114, USA

Abstract. A new method for the cooperative control of a system of multiple mobile robots with time delay in the network communication is presented. The network of mobile robots is modeled as a swarm of particles performing a directed random walk where the motion of the swarm is controlled by a central unit such as a robot leader. The collective motion of the robots is modeled by a system of stochastic delay-difference equations where the best solution found by the swarm is used as the network cooperative control signal. The method is applied to the solution of two problems. In the first problem, a group of autonomous underwater vehicles (AUVs) searches for the maximum depth in a two-dimensional domain. In the second problem, the group of AUVs searches for the minimum temperature in a three-dimensional domain. It is found that the search proceeds along Levy flights followed by sticking short random walks in the vicinity of the extremum points and that the cooperative control method is robust to time delays in network communication.

1 Introduction

In the past, robots have been used in many practical applications such as industrial robots in manufacturing, spacecraft and rover robots for space exploration and unmanned air vehicles (UAVs) for reconnaissance, surveillance and tactical military missions. Other possible applications include underwater missions by autonomous underwater vehicles (AUVs) such as formation control and rendezvous, search and rescue missions and exploration and mapping of unknown environments. At the beginning, single robots were employed in the performance of any given task. It has been recognized for some time, however, that the use of collaborating multiple mobile robots can have significant advantages in achieving complex tasks and missions which otherwise might not be achievable with single robots. Consequently, in recent years, researchers started treating remarkable problems related to the cooperative control of networked collaborating mobile robots with distributed resources such as sensors, computing power and communications [1], [2] and [3].

The present paper represents a step in that direction. Consider a specific implementation of a group of mobile robots in the form of a group of auto-nomous underwater vehicles, assigned with the collective task of exploring a limited domain of a body of water such as a lake, sea or a limited part of the ocean. Let's consider a specific task where the robots have to collectively find the extremum of an otherwise unknown

Crispin Y. (2005).
Cooperative Control Of A Robot Swarm With Network Communication Delay.
In *Proceedings of the 1st International Workshop on Multi-Agent Robotic Systems*, pages 3-14
DOI: 10.5220/0001193400030014

function using a given experimental method where they can measure a property of the environment such as water depth or temperature at any required point, see for example [4] and [5]. The simplest such test problem is for the group of robots to collectively search the maximum depth in a limited area; for example a group of 10 robots searching an area of 1 km by 1 km and measuring the depths using a depth finder device. Another problem might be to find the minimum temperature or maximum concentration of some chemical compound or density of plankton in a three dimensional domain, say a body of water of 1 km by 1 km by 100 m deep.

In developing the present method of cooperative control, it is assumed that each autonomous vehicle has a low level control system that can control the motion of the robot and bring it from one point in the domain to the next at the right speed and orientation. It is also assumed that each autonomous vehicle is equipped with a collision and obstacle avoidance control system for preventing collisions with other robots and obstacles. The robots network architecture is in the form of a leader robot acting as a server and communicating with the other robots as clients. The principle of control of the network of robots is the robot swarm cooperative control method described in the next section. Each robot has a microprocessor computing device on board, which is capable of running the robot swarm algorithm.

We propose to use the paradigm of a modified particle swarm optimization algorithm as a top level discrete event controller for the cooperative control of the group of mobile robots. Each robot sends the best solution found at any given time to the leader or other central processing station through its communication channel. The leader in turn computes the global best solution and transmits the result as a control signal to the network. The Particle Swarm Optimization (PSO) is a stochastic population based method that belongs to the class of biologically inspired algorithms. It is based on the paradigm of a swarm of insects performing a collaborative task such as ants or bees foraging for food using chemical or some other type of communication, see for example [6] and [7]. The method was originally developed by [8] and later described in great detail as a Swarm Intelligence method in [9]. An overview of the method as extensively applied to various function optimization problems of increasing difficulty, has recently been given by [10]. To the best of our knowledge, this is the first time that the PSO method has been modified and adapted for use as a top level discrete event cooperative control method for a swarm of autonomous robots. It is also the first time that the effect of information delay on the performance of the swarm has been studied in a PSO context.

In the next section we develop the robot swarm optimization algorithm with communication delay and we explain how it can be applied to solve the two problems mentioned previously. In the first problem, a group of robots collectively searches for the maximum depth in a limited 2-D domain. In the second problem, the group of robots searches for the minimum temperature in a 3-D domain.

2 Cooperative Control Of The Robot Swarm

The Robot Swarm cooperative control method is derived from the Particle Swarm Optimization algorithm. Two major modifications are made in order to implement the search method by actual mobile robots such as autonomous underwater vehicles or AUVs. The first modification imposes a limitation on the speed of the vehicle, or equivalently, a limit on the distance ΔX_{max} it can move in a time step Δt. The second modification takes into account the effect of imperfect communication between the group of robots. At any given time, communication with one or more robots can be completely cut off or otherwise attenuated or corrupted by noise due to the harsh underwater environment. Therefore, rather than assuming that the global minimum is available to the swarm at all times as in the case of perfect communication, we introduce a time delay in communicating the global minimum to all members of the swarm. To the best of our knowledge, the effect of a time delay on the performance of the swarm has not been treated so far in the literature. The particle swarm optimization algorithm with no speed constraints and with perfect communication consists of minimizing a function of several variables

$$\text{minimize } f(X), \text{where}$$

$$X \in \Omega \subset \mathbb{R}^n \text{ and } f : \Omega \mapsto \mathbb{R}$$

subject to the side constraints

$$X_{min} \leq X \leq X_{max}$$

using a directed random walk process described by the following system of stochastic difference equations:

$$X^i(k+1) = X^i(k) + V^i(k+1)\Delta t \tag{2.1}$$

$$V^i(k+1) = wV^i(k) + c_1 r_1^i (P^i(k) - X^i(k))/\Delta t +$$

$$+ c_2 r_2^i (P^g(k) - X^i(k))/\Delta t \tag{2.2}$$

Here w, c_1 and c_2 are real constants, $r_1^i$and r_2^i are random variables uniformly distributed between 0 and 1. The superscript index i denotes robot number $i \in [1, N_R]$ where N_R is the number of robots in the swarm and k is a discrete event counter. The velocity vector $V^i(k)$ has the same dimension n as the space position vector $X^i(k)$ and Δt is a typical time segment used to increment the motion of the swarm of robots in the domain Ω. Here $P^i(k)$ is the best solution found by robot i at time $t = k$ and $P^g(k)$ is the global minimum at time $t = k$.

This system of equations describes a directed random walk for each particle i in the swarm, similar to a Brownian motion of a tracer particle in a fluid. Whereas Brownian motion is an undirected random motion, the motion of a particle in the swarm will have a velocity that will start as a random motion, but will eventually decay as the particle approaches a point $P^{\,i}(k)$ in the domain where the function reaches a local minimum and as the swarm as a whole approaches a point $P^{\,g}(k)$ of the domain where the function reaches a global minimum, that is,

$$P^i(k) = \text{argmin} f(X^i(k))$$

$$P^g(k) = \text{argmin} f(P^i(k)),\ i \in [1, N_R] \tag{2.3}$$

The following initial conditions are needed in order to start the solution of the system of difference equations

$$X^i(0) = X_{min} + r^i \Delta X_{max} \tag{2.4}$$

$$V^i(0) = V_{min} + r^i \Delta X_{max}/\Delta t \tag{2.5}$$

$$\Delta X_{max} = (X_{max} - X_{min})/N_{\text{x}} \tag{2.6}$$

N_{x} is a typical number of grid segments along each coordinate component of the position vector X. For example, if the domain consists of a two dimensional square domain of 1000 m by 1000 m, then with $N_{\text{x}} = 40$, we can use a typical distance segment of $\Delta X_{max} = 1000$ m/$N_{\text{x}} = 25$ m. If we take a typical speed of an autonomous underwater robot as $V_c = 1$ m/s, then the typical time will be $t_c = \Delta X_{max}/V_c = 25$ s. We can now measure X in units of ΔX_{max}, V in units of V_c and Δt in units of t_c. The equations will then have exactly the same form in non-dimensional variables. We now modify the above algorithm such that it can be physically implemented by a group of cooperative underwater mobile robots. Before introducing the robot speed and communication constraints, we eliminate the time from the above equations by writing

$$\Delta X^i(k+1) = V^i(k+1)\Delta t \tag{2.7}$$

Then upon placing a limit on the magnitude of the velocity component of the robot in any given direction for a given time step , we can impose a constraint on the magnitude of the distance components in any given direction as

$$|\Delta X^i(k+1)| < |V_{max}|\Delta t = \Delta X_{max} \tag{2.8}$$

We introduce a time delay τ in the availability of the global minimum $P^g(k-\tau)$ at any given time k. The time delay will be on the order of 10 time steps or more. For example, this can be used to simulate imperfect communication between the robots in the group and a leader robot who receives $P^{\,i}(k)$ from all the robots in the group, computes the cooperative control signal $P^{\,g}(k)$ and sends the information back to the network. This can also simulate a communication failure between the leader and a small subgroup of robots for several time steps. We also make the velocity decay $w(k)$ factor time dependent to improve the search process when the global minimum is approached and smaller motion steps are needed for better resolution. Under these assumptions, the equations of motion of the swarm become:

$$X^i(k+1) = X^i(k) + sign(\Delta X^i(k+1))$$

$$(min[|\Delta X^i(k+1)|, \Delta X_{max}]) \tag{2.9}$$

$$\Delta X^i(k+1) = w(k)X^i(k) + c_1 r_1^i (P^i(k) - X^i(k)) +$$

$$+c_2 r_2^i (P^g(k-\tau) - X^i(k)) \tag{2.10}$$

subject to the side constraint

$$X_{min} \leq X^i(k+1) \leq X_{max} \tag{2.11}$$

The signum function term $sign(\Delta X^i(k+1))$ is added in order to keep the original direction of the motion while reducing the magnitude of the step. Suppose we would like to iterate the difference equations for N time steps, starting at a time $k = \tau + 1$. In this case the initial function is needed for the time interval $k \in [0, \tau]$. One possible method to simulate the delayed process is to start the iteration without any communication between the robots. This is a worst case scenario, and if it works, this will show that the swarm algorithm with delay is robust with respect to time delay in network communication. In this case the process can be started by replacing the global minimum $P^g(k)$ by the local minimum for each robot $P^i(k)$ in Eq.(2.10), i.e., there is no cooperative control and no communication over the initial time period $k \in [0, \tau]$.

$$X^i(k+1) = X^i(k) + sign(\Delta X^i(k+1))$$

$$(min[|\Delta X^i(k+1)|, \Delta X_{max}]) \tag{2.12}$$

$$\Delta X^i(k+1) = w(k)X^i(k) + c_1 r_1^i (P^i(k) - X^i(k)) +$$

$$+c_2 r_2^i (P^i(k) - X^i(k)) \tag{2.13}$$

for $k \in [0, \tau]$, with the initial condition:

$$X^i(0) = X_{min} + r^i \Delta X_{max} \tag{2.14}$$

This will generate $P^i(k)$ for the initial time segment $k \in [0, \tau]$. As the communication starts at time $k = \tau + 1$, we can obtain $P^g(k-\tau)$ from the best solution obtained at time $k = \tau$:

$$P^g(k-\tau) = \text{argmin} f(P^i(\tau)) \tag{2.15}$$

for $i \in [1, N_R]$. We can then iterate the equations of motion with delay for the time interval $k \in [\tau + 1, N]$.

$$X^i(k+1) = X^i(k) + sign(\Delta X^i(k+1))$$

$$(min[|\Delta X^i(k+1)|, \Delta X_{max}]) \tag{2.16}$$

$$\Delta X^i(k+1) = w(k)X^i(k) + c_1 r_1^i (P^i(k) - X^i(k)) +$$

$$+c_2 r_2^i (P^g(k-\tau) - X^i(k)) \qquad (2.17)$$

for the time interval $k \in [\tau + 1, N]$.

3 Collaborative Search In A 2-D Domain

In this section the cooperative control method developed in the previous section is applied to the problem of experimentally finding the minimum of a scalar function of two real variables. In the context of a group of underwater vehicles, the problem consists of finding the minimum of a scalar quantity such as depth, temperature, or the concentration of a chemical or biological species, through the measurement of the scalar quantity by the autonomous robots as they perform a search process in the domain. We would like to keep the robots resources to a minimum, so we limit the number of robots to 10, although we were able to minimize 2-D functions with as little as 6 robots.

Let's select a two-dimensional test function for which it is not easy to find the minimum, such as the banana function that has a curved valley.

$$f(X_1, X_2) = 10(X_1/d)^4 - 20(X_1/d)^2(X_2/d) +$$

$$+10(X_2/d)^2 + (X_1/d)^2 - 2(X_1/d) + 5 \qquad (3.1)$$

where $d = 200$ m. Consider a two dimensional domain of 1000 m by 1000m, defined by the coordinates:

$$X_1 \in [X_{1min}, X_{1max}] = [-500, 500]$$

$$X_2 \in [X_{2min}, X_{2max}] = [-500, 500] \qquad (3.2)$$

We choose the number of grid segments as $N_{\mathrm{x}} = 40$, so that the maximum distance traveled by any robot in any direction X_1or X_2 in one time step is 25 m, which we choose as one distance unit or 1 DU. The equivalent time unit TU is the time to travel along 1 DU at a typical speed of 1 m/s, i.e. 25 s.

$$\Delta t = 1TU = 25s$$

$$\Delta X_{1max} = \Delta X_{2max} =$$
$$= (X_{1max} - X_{1min})/N_{\mathrm{x}} = 25m = 1DU$$

$$|V_1|_{max} = |V_2|_{max} \leq \Delta X_{1max}/\Delta t =$$

$$= 1DU/1TU = 25m/1TU = 1m/s \qquad (3.3)$$

The following results are for $\tau = 20$ TU= 500 s. The number of autonomous robots is $N_R = 10$. The other parameters appearing in the equations of motion are

$c_1 = 1.5$ and $c_2 = 2.5$. $w(k)$ decreases from an initial value of $w_0 = 0.8$ to a final value of $w_f = 0.2$ after N steps:

$$w(k) = w_f + (w_0 - w_f)(N - k)/N \tag{3.4}$$

The results of a simulation of the 10 robots as they search for the minimum of the banana function are given in Figs. 1-3. The 10 robots were spread randomly over the domain at the start of the simulation, which was run over N= 120 steps. At the end of the simulation the trajectory of the robot that came closest to the location of the minimum at the point (X_1, X_2)=(200,200) was chosen for display. The velocity components are shown in Fig.1, with the limitation on the absolute values of the velocity components shown along some segments of the motion. The trajectory in parametric form, i.e., with the event counter k as a parameter, is given in Fig.2. Notice the long segments of motion, known as "Levy flights" with maximum speed along straight lines and segments where the robot performs a random walk about the same location. The trajectory is shown in Fig.3. The long Levy flights along straight lines are noticeable in the figures. Levy flights and anomalous diffusion occur in fluid flows, see for example [11], [12], [13], [14] and in other physical phenomena [15].

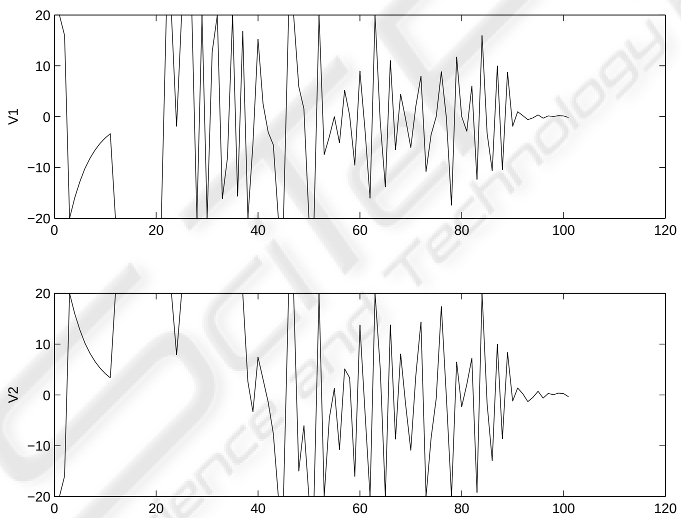

Fig. 1. Velocity components V_1 and V_2 in the X_1 and X_2 directions. The constraint on the maximum speed is apparent.

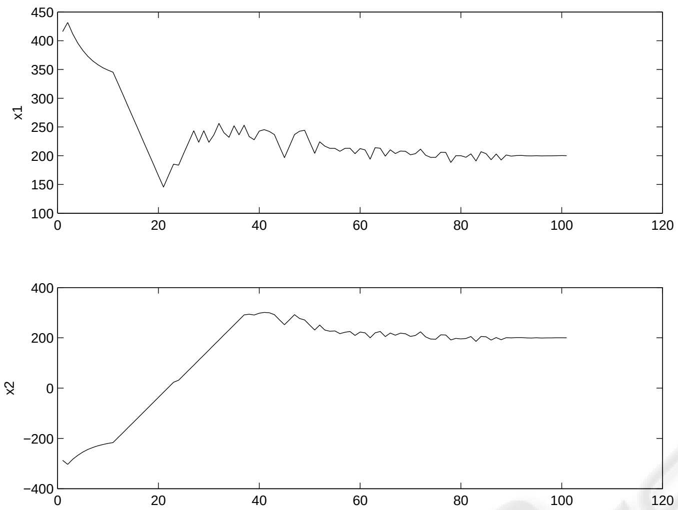

Fig. 2. Coordinates of the best trajectory X_1 and X_2 as a function of the event counter k. Levy flights are followed by sticking random walks.

4 Collaborative Search In A 3-D Domain

Here we consider a more difficult search problem, in which the swarm of robots is performing a search for the minimum of a scalar function of three real variables. In the context of autonomous underwater vehicles, the task here is to find a minimum temperature or maximum concentration of a chemical or biological species in a three-dimensional domain. The number of robots is limited to 10. We select a 3-D test function taken from the literature, for example the Levy No.8 function [10]. Consider a three-dimensional body of water with sides 1000 m by 1000 m by 1000 m deep. Select the origin of a cartesian system of coordinates in the center of this cube, such that the domain is defined by:

$$X_1 \in [X_{1min}, X_{1max}] = [-500, 500]$$

$$X_2 \in [X_{2min}, X_{2max}] = [-500, 500]$$

$$X_3 \in [X_{3min}, X_{3max}] = [-500, 500] \quad (4.1)$$

The Levy No.8 function is defined by

$$f(X_1, X_2, X_3) = sin^2(\pi y_1) + f_1(y_1, y_2) + \\ + f_2(y_2, y_3) + (y_3 - 1)^2 \quad (4.2)$$

$$f_1(y_1, y_2) = (y_1 - 1)^2(1 + 10 sin^2(\pi y_2))$$

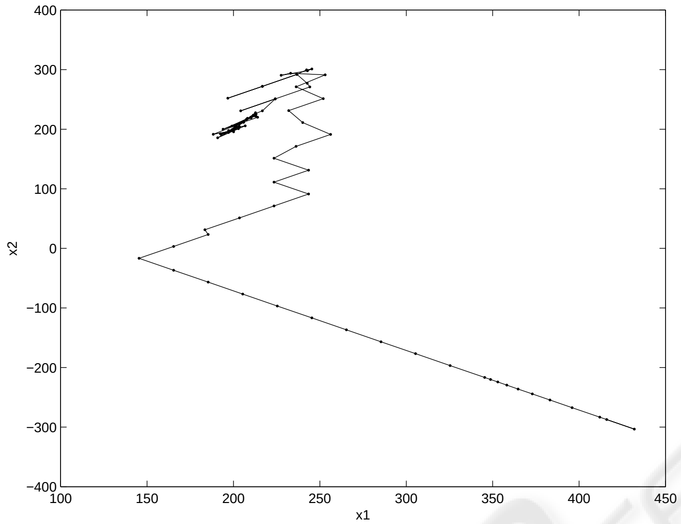

Fig. 3. Same trajectory as in Fig.2 with the starting point at the lower right corner and the end point at the minimum (200,200). Notice the Levy flights along straight line segments and random short walks around the minimum.

$$f_2(y_2, y_3) = (y_2 - 1)^2(1 + 10sin^2(\pi y_3)) \tag{4.3}$$

$$y_1 = 1 + (x_1 - 1)/4$$

$$y_2 = 1 + (x_2 - 1)/4$$

$$y_3 = 1 + (x_3 - 1)/4 \tag{4.4}$$

and the coordinates x_1, x_2, x_3 are scaled by a length $d = 50$ m:

$$x_1 = X_1/d, \ x_2 = X_2/d, \ x_3 = X_3/d \tag{4.5}$$

The results are for a delay of $\tau = 20$TU$= 500s$ and the number of autonomous robot vehicles is $N_R = 10$. The other parameters appearing in the equations of motion are the same as in the previous case of a 2-D function. The results of a simulation of the group of robots collaboratively searching for the minimum of the function of three variables are given in Figs. 4-6. The three velocity components are shown in Fig.4. The constraint on the absolute values of the velocity components is active along some segments of the motion. The three coordinates as a function of the event counter k are given in Fig.5. Again there are segments of long Levy flights at maximum speed along straight lines and segments where the robot performs a random walk in a limited area. The trajectory is shown in Fig.6 together with a contour plot of the projection of the f(X_1,X_2,X_3) function on the plane X_2=50. The minimum is located at (50,50,50). In the trajectory projection, it can be seen that segments of long Levy flights are followed by sticking random walks in the vicinity of the minimum.

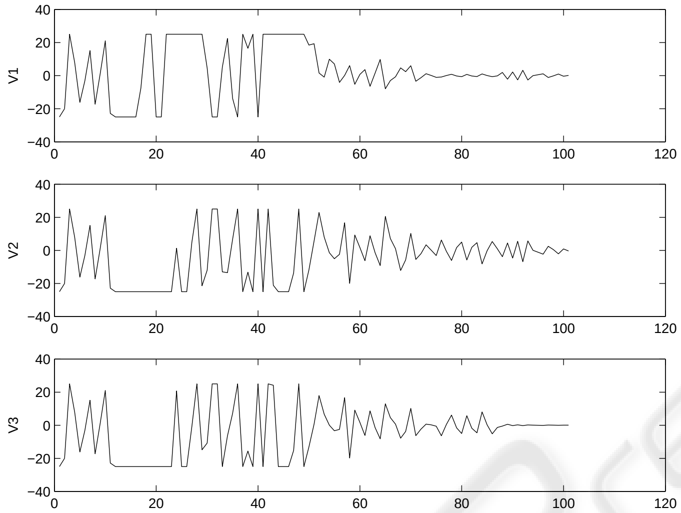

Fig. 4. Velocity components V_1, V_2 and V_3 in the X_1, X_2 and X_3 directions. The constraint on maximum speed is apparent.

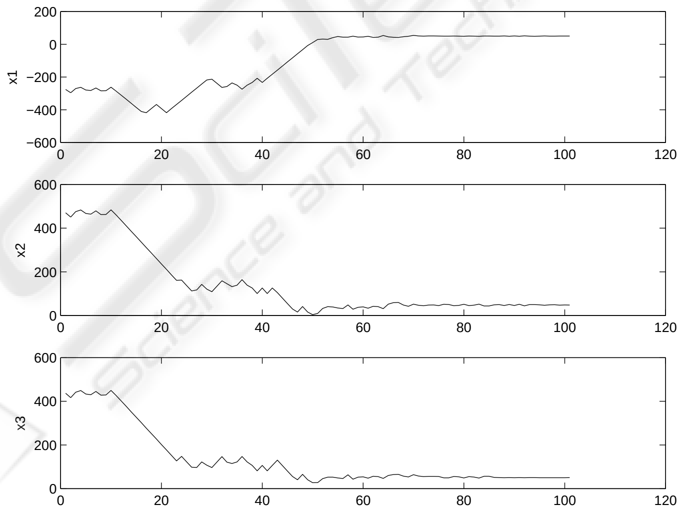

Fig. 5. Coordinates of the best trajectory X_1, X_2 and X_3 as a function of the event counter k. Long Levy flights are followed by sticking random walks in the vicinity of the minimum.

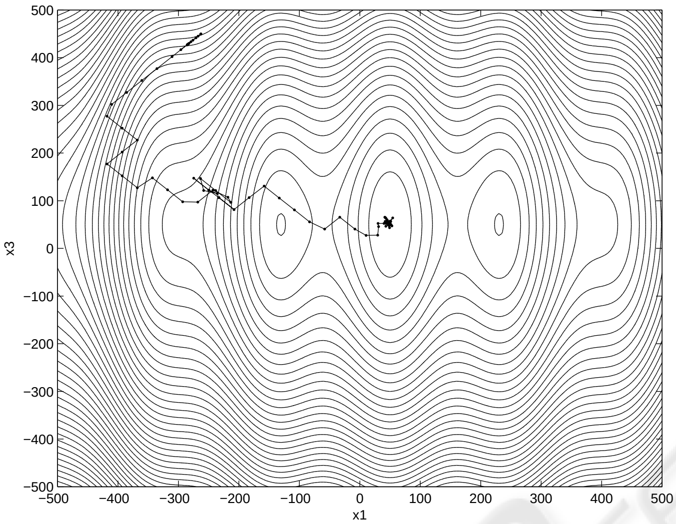

Fig. 6. Contour plot of the projection of $f(X_1,X_2,X_3)$ on the plane X_2=50, with the trajectory of one robot. The minimum is located at (50,50,50).

5 Conclusion

A method for the cooperative control of a group of robots based on a stochastic model of swarm intelligen ce has been developed. The network of mobile robots is modeled by a particle swarm moving randomly in the search domain with the global motion of the swarm directed and controlled by a central unit which can be a leader robot or a central server. The method takes into account time delays in the robots network communications. The motion of each robot in the swarm is governed by a system of two delay-difference equations. The best solution found collectively by the swarm serves as the control signal for the network of robots. The control signal can be time delayed due to communication failure with a subset of the robots in the swarm or due to noise or attenuation in any of the communication channels. The method was used to solve the basic problem of collaborative search and optimization in a 2-D and in a 3-D domain. It was found that the swarm can find the optimum despite long time delays in the network communications.

An unexpected result that was obtained is that the robots trajectories exhibit anomalous diffusion, performing long distance Levy flights along straight lines, followed by sticking random walks in a limited area of the domain, especially when the motion of the swarm starts converging to the location of the optimum.

References

1. DeSousa, J., Pereira, F.: On coordinated control strategies for networked dynamic control systems: an application to auvs. In: Proceedings of the 15th International Symposium of Mathematical Theory of Networks and Systems. (2002)

2. Sperazon, A.: On Control Under Com-munication Constraints in Autonomous Multi-Robot Systems. Licentiate Thesis, KTH Signals, Sensors and Systems, Royal Institute of Technology, Stockholm, Sweden (2004)
3. J. Silva, A. Speranzon, J.D., Johansson, K.: Hierarchical search strategy for a team of autonomous vehicles. In: IFAC. (2004)
4. E. Burian, D. Yoerger, A.B., Singh, H.: Gradient search with autonomous under-water vehicles using scalar measurements. In: IEEE Symposium on Autonomous Underwater Vehicle Technology, IEEE (1996)
5. Bachmayer, R., Leonard, N.: Vehicle networks for gradient descent in a sampled environment. In: Proceedings of the 41st IEEE Conference on Decision and Control, IEEE (2002)
6. E. Bonabeau, M.D., Theraulaz, G.: Swarm Intelligence, From Natural to Artificial Systems. Santa Fe Institute in the Sciences of Complexity, Oxford University Press, New York (1999)
7. M. Dorigo, G.D., Gambardella, L.: Ant algorithms for discrete optimization. In: Artificial Life, Vol.5, pp.137-172 (1999)
8. Eberhart, R., Kennedy, J.: A new optimizer using particle swarm theory. In: Proceedings of the 6th Symposium on Micro Machine and Human Science, IEEE Service Center (1995)
9. J. Kennedy, R.E., Shi, Y.: Swarm Intelligence. Morgan Kaufman Publishers, San Francisco (2001)
10. Parsopoulos, K., Vrahatis, M.: Recent approaches to global optimization problems through particle swarm optimization. In: Natural Computing, Vol.1, pp.235-306, 2002. (2002)
11. T.H. Solomon, E.W., Swinney, H.: Observation of anomalous diffusion and levy flights in a two-dimensional rotating flow. In: Physical Review Letters, Vol.71, p.3975, APS (1993)
12. T.H. Solomon, E.W., Swinney, H.: Chaotic advection in a two-dimensional flow: Levy flights and anomalous diffusion. In: Physica D, 76,pp.70-84. (1994)
13. E.R. Weeks, J.U., Swinney, H.: Anomalous diffusion in asymmetric random walks with a quasi-geostrophic flow example. In: Physica D, 97, pp.291-310. (1996)
14. Weeks, E., Swinney, H.: Anomalous diffusion resulting from strongly asymmetric random walks. In: Physical Review E, 57, 5, pp.4915-4920, APS (1998)
15. M.F. Schlesinger, G.Z., (eds), U.F.: Levy Flights and Related Topics in Physics. Springer-Verlag, Berlin (1995)