Proceedings of the 2007 IEEE/RSJ International
Conference on Intelligent Robots and Systems
San Diego, CA, USA, Oct 29 - Nov 2, 2007

Control of Robotic Swarm Behaviors Based on Smoothed Particle Hydrodynamics

Muhammed R. Pac, *Student Member*, *IEEE*,
Aydan M. Erkmen, *Senior Member*, *IEEE*, Ismet Erkmen, *Member*, *IEEE*

Abstract—**The paper presents a fluid dynamics based framework for control of emergent behaviors of robot swarms that are modeled as fluids. A distributed low-level control mechanism is developed based on Smoothed Particle Hydrodynamics (SPH) and it is coupled with a high-level control layer that is responsible for the control of fluid parameters to generate desired behaviors from the swarming characteristics of the robots. It is shown by simulations that using the same low-level SPH model, different swarming behaviors can emerge from the local interactions of robots according to the settings of the fluid parameters that are controlled by the high-level control layer.**

I. Introduction

SWARM robotics aims at developing scalable, flexible, and robust coordination mechanisms to control large groups of autonomous mobile agents. It is inspired by ethological phenomena in which swarms of animals (insects, fishes, birds, etc.) interact to coordinate their actions, create collective intelligence, and perform tasks that are far beyond the capabilities of individual members. Absence of central control in these behaviors and emergence of cooperation from local interactions makes social swarms highly fault-tolerant, scalable, and adaptive. It is these inherent properties of biological swarms, which are also desirable for collective robotics, that attract the interest of robotics research [1], [2]. Current approaches in the literature are generally based on the underlying phenomena of biological swarms and try to mimic the behaviors of animals with robots [3]–[5]. In these studies, adaptation of animal behaviors to multi-robot systems as a low-level coordination mechanism is mainly addressed. While constructing large numbers of autonomous robots is already a big challenge, developing control algorithms applicable to such systems based on the behaviors of animal swarms remains to be the main focus of the recent research.

In this paper, we propose a novel approach for the control of emergent and aggregate swarming behaviors of collective multi-robot systems. In contrast to the majority of approaches in the literature inspired by coordination in animal swarms, we base our formalism on the physics of fluids through an analogy between swarm robots and fluid particles. Our control methodology leads us to achieve desirable properties such as decentralized coordination, scalability, and robustness by applying the physical principles behind the dynamics of fluids to the distributed control of swarm robots. For the numerical analyses of the governing equations of fluid dynamics, we use *Smoothed Particle Hydrodynamics* (SPH) [6], which is a meshfree method that we found particularly suitable for our purposes of modeling a robot swarm as a collection of fluid particles.

Manuscript received April 9, 2007.

M. R. Pac is with the Space Technologies Research Institute of the Scientific and Technological Research Council of Turkey, Ankara, 06531 Turkey (phone: 00903122101310; fax: 00903122101315; e-mail: rasid.pac@uzay.tubitak.gov.tr).

A. M. Erkmen is with Middle East Technical University, Department of Electrical and Electronics Engineering, Ankara, 06531 Turkey (e-mail: aydan@metu.edu.tr).

I. Erkmen is with Middle East Technical University, Department of Electrical and Electronics Engineering, Ankara, 06531 Turkey (e-mail: erkmen@metu.edu.tr).

II. Previous Work

Apart from the popular "artificial potential field" approach to control of multi-robot systems [5], [7], there appears another line of research inspired from physics to solve problems such as coverage, surveillance, formation control, and obstacle avoidance. Spears and Spears propose a "physicomimetics" framework for distributed control of swarms of robots [8], [9]. In this framework, individual robots are treated as particles subject to artificial physics force laws. Mobile robots are driven by these virtual forces and eventually the system is expected to achieve a desired configuration that minimizes the overall system potential energy. Depending on the relative strengths of the attractive and repulsive forces between particles, the system acts like a solid, liquid, or gas. This approach was used to address the lattice formation problem. For the coverage problem, on the other hand, they proposed a "kinetic theory" approach in which the problem is handled as the sweeping of a corridor by a particle swarm [10].

Perkinson and Shafai [11] propose to control the positional organization and movement of a robot swarm based on SPH to address obstacle avoidance and coverage problems. To the best of our knowledge, this is the only work using SPH as a motion control algorithm for multi-robot systems. However, it does not establish an analogy between fluids and robot swarms and limits the method to the *self-deployment* of a sensor network.

The novelty in our previous works [12], [13] is not only to extend the idea of physics-based approaches by modeling a robot network as a fluid body but also to control the deployment process through the parameters available in the governing equations. In this paper, we further extend the previous formalism in [12] by developing a low-level, fluid dynamics based control model to coordinate the local interactions of robots while providing a set of flow parameters to higher level algorithms for controlling the global behavior of the system. We exploit SPH by synthesizing it with our control architecture through fluid dynamics equations. We demonstrate the validity and promise of the approach by applying it to common problems recurring in the swarm robotics literature.

III. PROPOSED CONTROL OF ROBOTIC SWARM BEHAVIORS

In this section, we present our motivation in developing a fluid dynamics based model for distributed control of robot swarms. There are various characteristics of fluids that are desirable in mobile robots, such as obstacle avoidance and source-to-sink optimal path finding behavior of fluids [14]–[17]. Similarly, harmonious and self-coordinated movement of fluids is also desirable in robot swarms. While obstacle avoidance is in part a local reactivity of the fluid, coherence of the whole body is a result of its aggregate state. It can be seen that these properties exist at conceptually two distinct scales: one is the *particle scale* that explains the local interactions of a fluid element with its surrounding, and the other is the *macroscopic scale* where the global motion of the fluid can be described. It is a fact that the overall motion of a fluid essentially emerges from local interactions of particles in it and the principles governing these interactions are based on various physical variables such as viscosity, compressibility, and temperature. Our idea in this paper is to utilize these variables to 'control' the swarming behaviors of a multi-robot system that we model as a collection of fluid particles with the mathematical formalism of SPH.

We developed a control architecture with two fundamental layers such that the lower layer deals with the particle scale of the swarm while the upper layer controls the macroscopic behavior. The two layers are in coordination through a set of fluid parameters that are described by the SPH model of the swarm in the lower layer. The global behavioral control of the swarm is designed in terms of the fluid parameters as a high-level control algorithm in the higher layer, whereas the local interactions of robots are governed by the low-level fluid model operating with this parameter setting. Fig. 1 depicts the relationships with bidirectional arrows between the control layers and interactions of robots with each other and with the environment. The interaction of a robot with its environment is basically for avoiding obstacles and interactions among robots are done through wireless communications. The high-level control layer is called the *Swarm Control Layer* that sets the flow parameters to appropriate values so that a particular swarm behavior emerges at the low-level control layer, called as the *SPH Layer*, that implements the desired swarm behaviors. Elements of this interface are of two types: those inherently available in the fluid dynamics equations and those belonging to an SPH-based swarm system. These parameters are summarized in Table I and discussed in the next section.

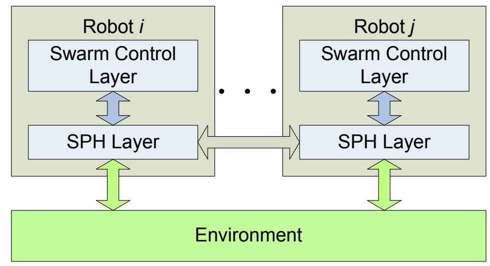

Fig. 1 Separate layers controlling the local interactions and global behavior in a decentralized system

TABLE I
FLUID MODEL PARAMETERS OF THE SPH LAYER

Parameter	Type	Effect
	Parameters from fluid physics	
Compressibility	Compressible	Gas-like swarms
	Incompressible	Liquid-like swarms
Viscosity (μ)	Viscous	Shear and normal stresses in robot-robot and robot-environment interactions
	Inviscid	No dragging stresses
Body force (f)	Available	Directional flow
	Not available	Self-spreading and flow
Gas Const. (R)	Gas-like swarms	Inter-particle repulsion
Initial density (ρ_0)	Liquid-like swarms	Inter-particle repulsion
Temperature (T)	Gas-like swarms	Obstacle avoidance
	Parameters of our SPH model of swarm robots	
Support Domain Size (R_d)	Deployment radius	Local interaction range
Kernel Function	Gaussian, spline, quadratic, etc.	Neighbor interactions
Connectivity parameters	Velocity damping factor	Constrains robot velocity to preserve connectivity
	Stiffness constant	Formation control of liquid-like swarms
Boundary parameter	Obstacle avoidance factor	Strength of the repulsive force normal from obstacles

IV. SPH FORMULATION FOR SWARM ROBOTS

A. Preliminaries

We present the SPH formulation in the most general form representing the flow of a *viscous* fluid in presence of *body forces* (e.g. gravitation). This form of the governing equations is called the *Navier-Stokes Equations* and is based on the conservation of three fundamental physical quantities: *mass*, *momentum*, and *energy*. For the details and derivations of these equations and related SPH formulations, the reader is referred to [18] and [6].

B. Definitions

Before introducing the SPH formulation, we present some basic definitions that will help clarify the adaptation of fluid dynamics concepts to robot swarms.

Particle: In SPH, a fluid body is represented by a collection of particles for each of which the governing equations of flow are 'independently' solved. Since, the Navier-Stokes Equations have no analytical solution, they are solved computationally in integral time steps –a technique called *time marching* [18] (pp. 85). In our framework, a particle corresponds to a mobile robot while the swarm corresponds to the whole fluid body.

Support Domain: Each particle in SPH has a support domain Ω, a set of neighboring particles within its locality. All calculations for each particle are carried out over its support domain at marching time instants. For the members of a swarm, this concept is equivalent to a *deployment neighborhood* of a robot, a set of neighboring robots inside a certain radius (R_d), which is less than or equal to the maximum communication radius (R_c) of that robot.

Smoothing (Kernel) Function: The state of a fluid is represented by a set of particles in SPH. Numerical discretization is made by approximating the values of field functions, their derivatives, or integrals at particle locations where neighboring particles contribute to the particle approximations based on their influence on the location. It is the smoothing function that determines the values of these contributions. The SPH formalism starts with:

$$f(\boldsymbol{x}) = \int_{\Omega} f(\boldsymbol{x}')\delta(\boldsymbol{x}-\boldsymbol{x}')d\boldsymbol{x}' \qquad (1)$$

where f is a field function, e.g. pressure, and Ω is the domain of the integral containing the position vector $\boldsymbol{x}$. As an approximation to the above integral representation, Dirac delta function δ is replaced with a smoothing kernel and the integral is changed to a finite summation over the support domain as in (2) where W is the smoothing function and V_j is the volume attributed to particle j. The angle brackets designate that the representation is an approximation.

$$< f(\boldsymbol{x}_i) > \; = \sum_{j\in\Omega_i} f(\boldsymbol{x}_j)W_{ij}(\boldsymbol{x}_i-\boldsymbol{x}_j)V_j \qquad (2)$$

In our formalism, we use the *Gaussian* kernel (Fig. 2) for its closed form expression and accuracy in disordered particles [6] (pp. 63). The kernel is given in (3), where R is a scaled distance between the particle for which the kernel is being computed and its neighbors in the support domain. The *smoothing length h* defines the influence area of the smoothing function.

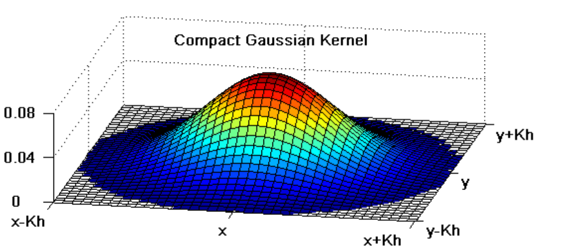

Fig. 2 The Gaussian kernel over a support domain centered at (x,y)

$$W(R_{ij}) = \begin{cases} \alpha_d e^{-R_{ij}^2} & R_{ij} \le R_d = \kappa h \\ 0 & otherwise \end{cases} \qquad (3)$$

$$\alpha_d = 1/\pi h^2 \quad R_{ij} = \left|\boldsymbol{x}_i - \boldsymbol{x}_j\right|/h \quad \kappa = 2$$

C. The SPH Formulation

In our SPH formalism, each robot in the swarm represents a computational element and solves the *momentum equation* to determine its acceleration based on the information obtained from neighbors in its support domain. Hence, the computational method is naturally distributed and parallel. Particle approximations for the fluid dynamics equations of density and momentum in (4) to (6) are adapted from [6] (pp. 113–123). In these equations, ρ stands for *density*, m for *mass*, and $\boldsymbol{v} = (u, v)$ for the *velocity* of particle i. Density at the location of a robot is formulated by the weighted sum of the neighbor masses as in (4), where W_{ij} is the kernel evaluated for robot i and its neighbor j. In (5), $\boldsymbol{f} = (f^x, f^y)$ is the *body force* acting on each robot. In fluid physics, a typical body force is the gravitation, whereas the swarm control layer of our approach uses it to impose a direction of flow whenever relevant to the desired swarming behavior. *D/Dt* is the *total derivative* operator with respect to *time*. σ is the *total stress tensor* given by (6), where p is the *pressure* and μ is the viscosity. Some of the parameters like mass and viscosity in these equations are only virtual properties for robots and are used without redefinition to obtain the natural characteristics of fluids in our model. However, all of the parameters involved in these equations can be adjusted by the swarm control layer to generate a desired swarming behavior.

The total stress tensor σ describes the interactions among particles. For instance, pressure is a potential field in a global view and produces inter-particle repulsion that results in flow toward homogeneous distribution. It is exactly the same effect of pressure that we employ in our approach to control the uniform distribution of the swarm. Viscous stress terms multiplied by μ in (6) regulate the velocity field by producing inter-particle drag forces. In our approach, these are analogous to attractive forces around a robot such that each neighbor in the vicinity imposes its own velocity on that robot. Therefore, total stress tensor is one of the main driving entities for swarm flow and harmony among robots.

$$\rho_i = \sum_{j\in\Omega_i} m_j W_{ij} \qquad (4)$$

$$\begin{aligned} \frac{Du_i}{Dt} &= \sum_{j\in\Omega_i} m_j \left(\frac{\sigma_i^{xx}+\sigma_j^{xx}}{\rho_i\rho_j}\frac{\partial W_{ij}}{\partial x_i} + \frac{\sigma_i^{xy}+\sigma_j^{xy}}{\rho_i\rho_j}\frac{\partial W_{ij}}{\partial y_i} \right) + f_i^x \\ \frac{Dv_i}{Dt} &= \sum_{j\in\Omega_i} m_j \left(\frac{\sigma_i^{xy}+\sigma_j^{xy}}{\rho_i\rho_j}\frac{\partial W_{ij}}{\partial x_i} + \frac{\sigma_i^{yy}+\sigma_j^{yy}}{\rho_i\rho_j}\frac{\partial W_{ij}}{\partial y_i} \right) + f_i^y \end{aligned} \qquad (5)$$

$$\sigma_i^{xx} = -p_i + \mu_i\left(2\frac{\partial u_i}{\partial x} - \frac{2}{3}\left(\frac{\partial u_i}{\partial x} + \frac{\partial v_i}{\partial y}\right)\right)$$
$$\sigma_i^{xy} = \mu_i\left(\frac{\partial v_i}{\partial x} + \frac{\partial u_i}{\partial y}\right) \qquad (6)$$
$$\sigma_i^{yy} = -p_i + \mu_i\left(2\frac{\partial v_i}{\partial y} - \frac{2}{3}\left(\frac{\partial u_i}{\partial x} + \frac{\partial v_i}{\partial y}\right)\right)$$

In (5), u_i and v_i are velocity components of robot i in x and y directions, respectively, and are what each robot ultimately needs to find out to control its motion. As for the pressure, there are two definitions according to the fluid being compressible (gas) or incompressible (liquid). The compressible case is modeled as in (7) where pressure is a function of density through the *state equation of gases* [18] (pp. 79). R and T are the specific gas constant and the absolute temperature, respectively. Equation (7) is the statement of the fact that pressure is linearly related with density within the support domain of each robot where increased number of neighboring robots results in an increase in the pressure at that particular location. We use R as a swarm specific parameter to adjust the inter-robot repulsion. In obstacle-laden environments, for example, R can be reduced to allow close spacing of robots without producing excessive pressure while moving across narrow passages. T, on the other hand, is used as a secondary mechanism of obstacle avoidance (the primary mechanism is the boundary condition described in part D) in a way that when a robot encounters an obstacle, it raises its temperature and produces higher pressure at its location so that its neighbors are repelled from the obstacle without actually coming across with it.

For incompressible flow, due to inefficiencies in solving for the actual state equation of liquids, the artificial compressibility concept is used to practically model incompressible fluids [6] (pp. 136). We formulate incompressible flow by using the *artificial compressibility* definition of [19] as given in (8) where ρ_0 stands for the initial density of the liquid and k is the *stiffness* constant. The greater k is, the more accurate incompressibility is simulated but in expense of smaller time steps for processing. For a wireless swarm system, therefore, the value of k is limited with the bandwidth of local communication among robots because each robot needs to communicate with its neighbors to collect flow information and process its equations at each time step.

$$\text{Compressible Flow:} \quad p_i = \rho_i R_i T_i \qquad (7)$$

$$\text{Incompressible Flow:} \quad p_i = k\left(\rho_i - \rho_0\right) \qquad (8)$$

The apparent difference between compressible and incompressible flow benefits to different application scenarios in a swarm system. For example, compressible flow is suitable for tasks that require the coverage of an area because the self-spreading behavior of gases is desirable in a mobile surveillance system. On the other hand, incompressible flow is more appropriate for patrolling and flocking tasks that require stiff formations among robots along a track.

D. Motion Control with Momentum Equation

As shown in the previous equations, each robot needs to communicate with its deployment neighbors to get their flow variables, such as density and velocity, and use them in its own flow calculations so that it can solve the momentum equation in (5) to find out its acceleration and control its motion. The solution is obtained using the time-marching technique that evolves the value of velocity by the first-order Taylor series approximation in (9).

$$\boldsymbol{v}_i^{t+\Delta t} = \boldsymbol{v}_i^{t} + \left(\frac{D\boldsymbol{v}_i^t}{Dt}\right)\Delta t \qquad (9)$$

Each robot controls its velocity based on this solution at incremental time instants as it moves in the environment. Since we assume robots as point particles, we do not deal with any dynamics of robots in this work. Depending on the application and the type of robots, it is the particular motion controller of the robot that accepts the velocity controls in (9) and deals with the physical dynamics.

Despite the very same equations are employed, different flow patterns are observed in different environments due to varying boundary conditions that are imposed by the environment surrounding the robots. Thus, sources of the robot-environment interactions are obstacles within the deployment terrain. The characteristic obstacle-avoidance behavior of fluids is modeled in our approach through the associated boundary conditions. For inviscid flow ($\mu = 0$), the only boundary condition is that the velocity of a robot adjacent to a surface is parallel to it. For viscous flow ($\mu > 0$), there is an additional *no-slip* condition such that the velocity of the robot reduces to zero at the immediate vicinity of the surface. We utilize an obstacle avoidance mechanism to satisfy these conditions as follows. First, we assume that the information obtained from the sensors of a robot basically carries relative range and bearing data of the obstacle. Also, if the robot has multiple sensors around its perimeter or a scanning detector, it is highly probable that it detects more than one point of an obstacle as illustrated in Fig. 3. Then, it can reason about a surface and adjust its velocity according to the boundary condition such that the velocity component perpendicular to the surface is decreased with decreasing distance to the obstacle. If the robot happens to detect only one obstacle point, then it can assume that the surface normal originates from this point and passes through its own location. Equation (10) briefly formulates the boundary condition such that the velocity component perpendicular to the surface decreases with decreasing k value. The parallel component of velocity $\boldsymbol{v}_p$ to the surface can be derived from the three locations ($\boldsymbol{x}_0$, $\boldsymbol{x}_1$, and $\boldsymbol{x}_2$) in Fig. 3. In the expression of k below, R_s represents the sensing range of the robot.

$$\boldsymbol{v}' = k\boldsymbol{v} + (1-k)\boldsymbol{v}_p \quad , \quad k = d/(2R_S - d) \tag{10}$$

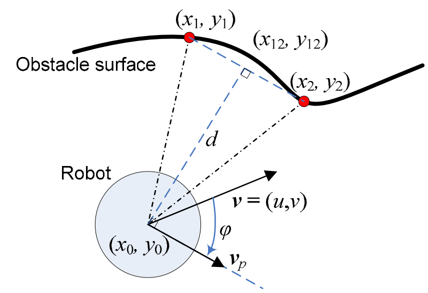

Fig. 3 Obstacle avoidance through boundary conditions

There are also some conditions originating from the nature of swarm robotics. For example, the members of a swarm have limited mobility capabilities which may not fully satisfy the direct solution obtained from flow equations. In our model, robot velocities are clipped by hard-limiters whenever the flow solution exceeds these limits. Robots also need to preserve the wireless connectivity among each other so that no robot detaches from the swarm. We use a *damping factor* to slow down the outliers in the swarm and also utilize viscous stress among robots to satisfy this connectivity constraint.

V. SPH Based Swarm Characteristic

In this section, we present sample simulation results to address some common problems in swarm robotics literature in order to demonstrate the capabilities and potential use of our SPH based control strategy.

A. Deployment and Coverage

The first simulation aims to analyze the effect of viscosity (μ) on coverage tasks of mobile sensor networks as it is varied as a model parameter. We simulate the compressible fluid model of a robot swarm in a corridor sweeping operation (Fig. 4). The gas-like swarm model is particularly suitable for coverage tasks due to the self-spreading nature of gases. We use the body force parameter $\boldsymbol{f}$ set to 0.5 Newton toward right so that the robots released from the left hand side of the corridor sweep through to the right while covering spare areas and avoiding obstacles. Introduction of viscosity into the flow is expected to result in frictional loss and hence in a slower movement especially around obstacles.

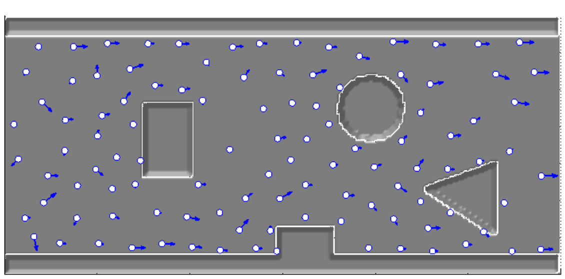

Fig. 4 Flow of robots from left to right in an obstacle-laden corridor

The plot in Fig. 5 typically shows that the mean coverage is lower when viscosity is nonzero. From Fig. 6, it is seen that the average velocity of the robots is also reduced under friction. However, reduction in coverage can better be explained with the increased average robot density when viscosity is nonzero as shown in Fig. 7. Therefore, it can be concluded that increased viscosity results in slower deployment, whereas it increases the average density of the robots. Although this is not the primary mechanism of controlling the overall density of the robots in the swarm, which is discussed in the last simulation, it can be utilized as a mechanism of slowing the movement down around obstacles for safe wandering and around points of interest in the terrain that require more careful inspection and are virtually assumed as obstacles by the robots.

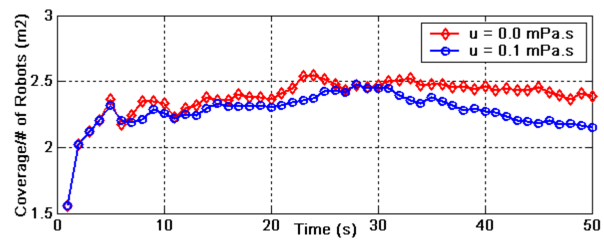

Fig. 5 A typical comparison of mean coverage for inviscid (μ = 0.0 mPa.s) and viscous flows (μ = 0.1 mPa.s)

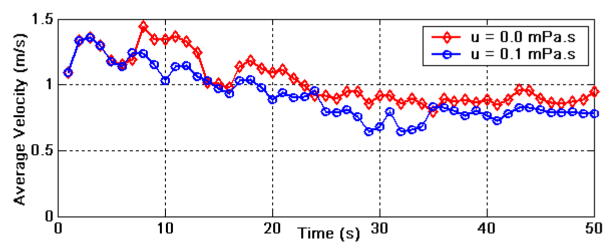

Fig. 6 Average velocities of robots under zero and nonzero viscosity

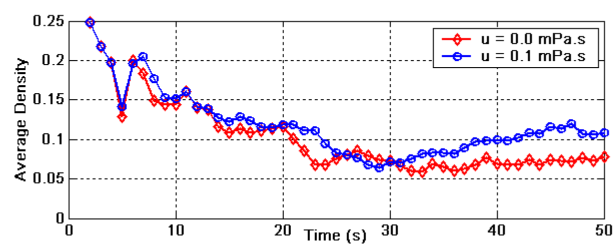

Fig. 7 Average densities of robots under zero and nonzero viscosity

B. Patrolling and Dispatching

In our SPH model, there are several flow parameters suitable for utilization in tasks, such as security patrolling and vehicle dispatching, which require a guided motion of a swarm of robots toward a target along a predefined route. Body force $\boldsymbol{f}$ is the parameter that plays the most important role in this respect. In the previous corridor sweeping example, body force was constant in magnitude and direction and was used to direct the swarm to the right of the corridor. For a patrolling or dispatching task, however, a position-varying body force has to be used. An example to such tasks is the dispatching of autonomous ground vehicles (AGV) in outdoor terrains based on a predetermined route and GPS data. Also, the compressibility parameter adjusted for incompressible flow becomes appropriate to gather and funnel down the robots along a specified route since liquids do not spread out as gases do.

Fig. 8 shows an autonomous dispatching scenario, where robots are given 5 waypoints by the swarm control layer in the terrain to navigate through. Upon arrival to a waypoint, each robot updates the body force guiding its motion to head toward the next waypoint. It is shown in this simulation that for tasks requiring a directed movement of the swarm in unstructured terrains, where obstacles are identified based on the variation of the steepness of the terrain, utilizing a dynamically changing body force parameter is effective. Moreover, modeling the swarm as a liquid rather than a gas is particularly beneficial in keeping the robots on a thin track.

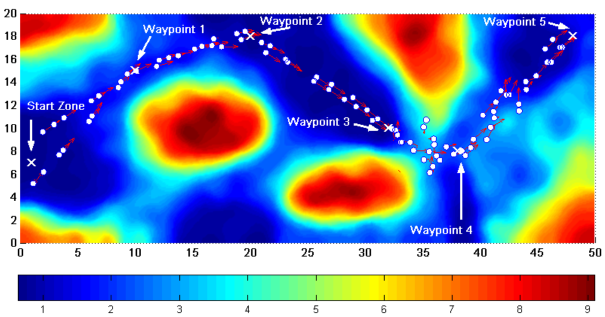

Fig. 8 Dispatching of robots through waypoints in a rural terrain

C. Flocking and Formation Control

Flocking is the synchronized and harmonious movement of a collection of agents as a single body –a behavior demonstrated by flocks of birds, schools of fishes, etc. While it has been modeled as an emergent distributed behavior [20] inspiring from flocks of animals, flocking is inherently available in the nature of fluids and equivalently represented in the governing equations by the total stress tensor in (6).

As an example to the control of the flocking behavior, we consider highly viscous incompressible flow, in which high viscosity results in strong cohesion among particles. We propose this form of fluid flow especially for *leader-follower* type flocking behavior, in which one or multiple leader robots in the swarm govern the overall movement through local interactions. The guiding effect of the leader in the swarm is analogous to a *point force* called *Stokeslet* [21] (pp. 450) applied on a particle in the fluid. Such a force induces a velocity field that radially propagates around this particle as in Fig. 9.

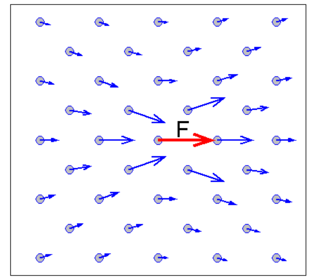

Fig. 9 Velocity field induced around a point force

Formation control is one of the most studied behaviors relevant to multi-robot systems and the very basic feature of a formation is the spacing between robots. The parameter in Table I that primarily affects the inter-robot distances of the swarm is the support domain radius R_d, which is limited by the communication range (R_c) of the robots. In the following simulation, we show the effect of this parameter on the spacing between robots using a compressible fluid model and assume that R_c is 2m for each robot. Fig. 10 shows a compact initial distribution of robots at the start of the simulation. We run the simulation for two different values of R_d, 1m and 1.6m. The final distribution of the robots reached after spreading out and stopping due to the connectivity constraint is given in Fig. 11. It is seen that the separation among robots is larger when R_d is increased. Fig. 12 also shows that when R_d is changed from 1m to 1.6m, the average separation among neighboring robots increases from 1.43m to 1.75m. For both cases, the standard deviation around the average separation is less than 3% after the swarm ceases to move.

VI. Conclusion

We presented a novel approach to the control of robotic swarm behaviors based on Smoothed Particle Hydrodynamics by modeling a robot swarm as a fluid and controlling its flow through the flow parameters involved. We showed here a control architecture that can shape emergent behaviors of robot swarms by controlling both local interactions of robots and the global behavior of the whole swarm system. The theory of fluid dynamics and the associated numerical analyses techniques like SPH are mature and provide a profound background to its exploitation in our framework. Due to the limited space in this paper, we were unable to present extensive simulation results that demonstrate the individual effect of each fluid parameter in Table I and its potential utilization in a swarming task [22]. Yet, the present discussion reveals the potential applicability of the model to various swarm robotics problems and is considered a precursor for the development of a fully explored fluid dynamics model for collective mobile robot systems.

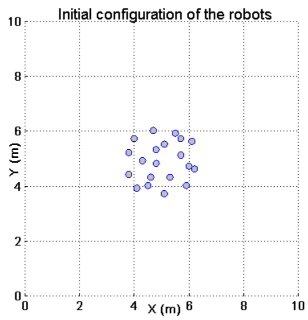

Fig. 10 Compact initial distribution of robots at t = 0

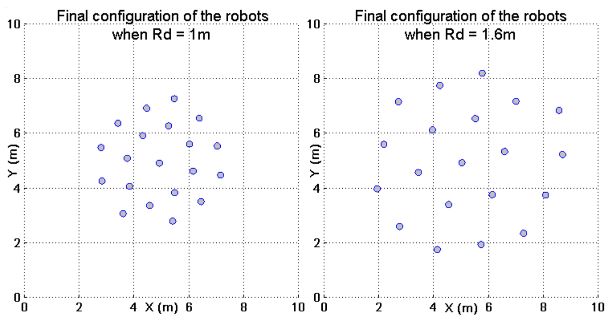

Fig. 11 Final distribution of robots at t=20s for two values of R_d

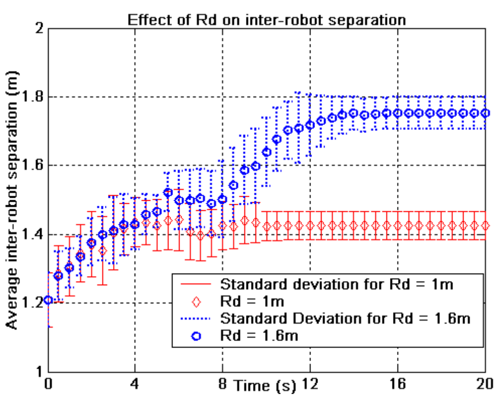

Fig. 12 Average inter-robot separation for R_d = 1m and 1.6m

REFERENCES

[1] G. Beni, "From Swarm Intelligence to Swarm Robotics," in *Swarm Robotics: State-of-the-art Survey*, *Lecture Notes in Computer Science 3342*, pp. 1–9, E. Şahin, W. M. Spears, Eds., Springer-Verlag, 2005.

[2] E. Şahin, "Swarm Robotics: From Sources of Inspiration to Domains of Application," in *Swarm Robotics: State-of-the-art Survey*, *Lecture Notes in Computer Science 3342*, pp. 10–20, E. Şahin, W. M. Spears, Eds., Springer-Verlag, 2005.

[3] T. Balch, "Communication, Diversity and Learning: Cornerstones of Swarm Behavior," in *Swarm Robotics: State-of-the-art Survey*, *Lecture Notes in Computer Science 3342*, pp. 21–30, E. Şahin, W. M. Spears, Eds., Springer-Verlag, 2005.

[4] F. Mondada, G. C. Pettinaro, A. Guignard, I. W. Kwee, D. Floreano, J-L. Deneubourg, S. Nolfi, L. M. Gambardella, M. Dorigo, "Swarm-Bot: A New Distributed Robotic Concept," *Autonomous Robots* 17, pp. 193-221, 2004.

[5] D. H. Kim, H. Wang, S. Shin, "Decentralized Control of Autonomous Swarm Systems Using Artificial Potential Functions: Analytical Design Guidelines," in *Journal of Intelligent Robots and Systems*, vol. 45(4), pp. 369–394, 2006.

[6] G. R. Liu and M. B. Liu, *Smoothed Particle Hydrodynamics: A Meshfree Particle Method*, World Scientific, 2003.

[7] A. Howard, M. J. Mataric, G. S. Sukhatme, "Mobile Sensor Network Deployment using Potential Fields: A Distributed, Scalable Solution to the Area Coverage Problem," in *Proc. of the 6th Int. Symp. on Distributed Autonomous Robotics Systems*, Fukuoka, Japan, June 25-27, 2002, pp.299 – 308.

[8] W. M. Spears, D. F. Spears, J. C. Hamann, and R. Heil, "Distributed, Physics-Based Control of Swarms of Vehicles," in *Autonomous Robots* 17, pp. 137-162, Kluwer Academic, 2004.

[9] W. M. Spears, D. F. Spears, R. Heil, W. Kerr, and S. Hettiarachchi, "An Overview of Physicomimetics," in *Swarm Robotics: State-of-the-art Survey*, *Lecture Notes in Computer Science 3342*, pp. 84–97, E. Şahin, W. M. Spears, Eds., Springer-Verlag, 2005.

[10] D. Spears, W. Kerr, and W. Spears, "Physics-Based Robot Swarms For Coverage Problems," in *International Journal on Intelligent Control and Systems*, vol. 11(3), 2006.

[11] J. R. Perkinson and B. Shafai, "A Decentralized Control Algorithm for Scalable Robotic Swarms Based on Mesh-Free Particle Hydrodynamics," in *Proc. of the IASTED Int. Conf. on Robotics and Applications*, Cambridge, USA, Oct. 31- Nov. 2, 2005.

[12] M. R. Pac, A. M. Erkmen, and I. Erkmen, "Scalable Self-Deployment of Mobile Sensor Networks: A Fluid Dynamics Approach," in *Proc. of the 2006 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Beijing, China, Sep. 9-15, 2006, pp. 1446–1451.

[13] M. R. Pac, A. M. Erkmen, and I. Erkmen, "Towards Fluent Sensor Networks: A Scalable and Robust Self-Deployment Approach," in *Proc. of the 1st NASA/ESA Conf. on Adaptive Hardware and Systems*, Istanbul, Turkey, June 15-18, 2006, pp. 365–372.

[14] C. Louste and A. Liegeois, "Near optimal robust path planning for mobile robots: the viscous fluid method with friction," in *Journal of Intelligent and Robotic Systems*, vol. 27(1-2), pp. 99–112, Jan. 2000.

[15] M. Kazemi, M. Mehrandezh, and K. Gupta, "An Incremental Harmonic Function-based Probabilistic Roadmap Approach to Robot Path Planning," in *Proc. of the 2005 IEEE Int. Conf. on Robotics and Automation*, Barcelona, Spain, April 2005, pp. 2136–2141.

[16] D. Keymeulen and J. Decuyper, "Self-organizing System for the Motion Planning of Mobile Robots," in *Proc. of the 1996 IEEE Int. Conf. on Robotics and Automation*, Minneapolis, Minnesota, April 1996, pp. 3369–3374.

[17] J.-O. Kim and P. K. Khosla, "Real-Time Obstacle Avoidance Using Harmonic Potential Functions," in *IEEE Transactions on Robotics and Automation*, vol. 8(3), pp. 338–349, June 1992.

[18] J. Anderson, *Computational Fluid Dynamics*, McGraw-Hill, 1995.

[19] M. Desbrun and M.-P. Cani, "Smoothed Particles: A new paradigm for animating highly deformable bodies," in *Proc. of Eurographics Workshop on Computer Animation and Simulation*, Aug. 1996, pp.61–76.

[20] C. W. Reynolds, "Flocks, Herds, and Schools: A Distributed Behavioral Model," *Computer Graphics*, 21(4), pp. 25-34, July, 1987.

[21] C. Pozrikidis, *Fluid Dynamics: Theory, Computation, and Numerical Simulation*, Kluwer Academic Publishers, 2001.

[22] M. R. Pac, "A Fluid Dynamics Framework for Control of Mobile Robot Networks," *M.S. Thesis*, Dept. Electrical and Electronics Eng., Middle East Technical Univ., Ankara, Turkey, 2007.