Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

Optimal Dynamic Actuator Location in Distributed Feedback Control of A Diffusion Process

YangQuan Chen, Zhongmin Wang and Jinsong Liang
†Center for Self-Organizing and Intelligent Systems (CSOIS)
Dept. of Electrical and Computer Engineering
4160 Old Main Hill, Utah State University, Logan, UT 84322-4160, USA

***Abstract*— In this paper, the problem of distributed neutralization of toxic 2D diffusion process is discussed. The diffusion process is modelled by a parabolic PDE system. A group of mobile robots which can release neutralizing chemicals are sent to detoxify the pollution. We attempt to solve the optimal actuator location problem by using Centroidal Voronoi Tessellations. A new simulation platform (`Diff-MAS2D`) for measurement scheduling and controls in distributed parameter systems is also introduced in this paper. Simulation results show the effectiveness of our proposed method.**

***Index Terms*— Diffusion process, pollution neutralization, Centroidal Voronoi Tessellation, distributed parameter system simulation.**

I. Introduction

With the advancement on large-scale integrated circuit design, reliable and energy-efficient wireless communication, the deployment of many small mobile robots on a vast area for spatial environment monitoring and control become more realistic. Mobile Actuator/Sensor Network (MAS-net) project developed at CSOIS, Utah State University, is a project that uses the small-scale mobile robots for the spatially distributed diffusion process monitoring and control [1], [2], [3]. This project combines mobile robotics with the wireless sensor networks. Each robot has limited sensing ability and limited communication ability. They are expected to coordinate with each other to control the diffusing process by temporal-spatial feedback closed-loop control. The application of this project can be in homeland security, where chemical, biological, radiological or nuclear (CBRN) terrorism can cause devastating damages. It is thus important to have a system that can respond and control the diffusion process of the harmful materials. Some research challenges and opportunities are presented in [4].

In this paper, we are considering the pollution neutralization problem in MAS-net project. The scenario is described as follows: A toxic diffusion source is releasing toxic material in 2D plane. The diffusion process is modelled as a parabolic PDE system. Chemical concentration sensors are deployed to cover the polluted area and collect data about the pollution. Then, a few of mobile robots equipped with controllable dispensers of neutralizing chemicals are sent to the polluted area with the mission to eliminate the pollution by properly releasing the neutralizing chemicals. In this case, the mobile robots would be autonomous boats. This paper tries to solve the problem how to choose the optimal positions for these robots and the trajectories the robots will follow when the dynamic diffusion is evolving.

What we most concern here is the minimal impact to the natural environment for both the pollution and neutralization process. The pollution can have severe negative impact on the the natural environment. It is expected to be removed as quickly as possible. But the neutralizing chemicals may also have negative impact on the natural environment, such as in acid-alkaline neutralization. If the strategy is too aggressive and too much or too fast the neutralizing chemicals are released, some part or even the whole area concerned may be negatively affected. This is not beneficial to maintain the healthy environment there. So, one of our objective is to deploy the robots and control the releasing process in an optimal way to minimize any negative impact on the environment. On the other hand, the neutralizing chemicals should be released in such a way that the diffusion of the pollution is bounded so that the heavily affected area is kept as small as possible. These all depend on how the robots positions are chosen, how they move and what the control law is to release the neutralizing chemical. Therefore, the problem of area coverage, robot motion planning and the dynamic diffusion process are fundamentally interrelated.

Our research is related to other research topics, for example, the coverage problem [5], the sensor deployment problem [6], [7] and the feedback control in PDE systems [8], [9]. In [5], the problem of complete coverage of the free space by a team of mobile robots is discussed. The free space is decomposed into cells. Each robot has onboard sensors and limited communication ability. A coverage algorithm has been developed to ensure that every cell is visited. Potential field method is used in [10] to produce force so that each senor is repelled by both the obstacles and other sensors. The sensors are expected to spread out to cover the whole area. In [5], [10], the whole area is treated equally without any differentiation. In [11], a decentralized gradient-search-based algorithm was proposed for motion planning of mobile sensors to improve the performance of estimating the states of a dynamic target. What discussed in [11] is closely related to our topic in this paper, but the motion planning of actuators in a PDE system for feedback control remains an open question.

Motivated by the application of Centroidal Voronoi Tessellation (CVT) in optimal placement of resource [12] and in coverage control of mobile sensing networks [6], we proposed a practical algorithm based on CVT to solve the problem of actuator motion planning to neutralize the pollution. An application of CVT in feedback control system can be found in [9]. In [9], the sensor location problem in feedback control of partial differential equation system is solved by CVT. The functional gains are served as the density functions in CVT. In our experiment, the pollution concentration is given by the sensors that cover the area and form a mesh. A simulation platform called `Diff-MAS2D` [13] has been developed for measurement scheduling and controls in distributed parameter systems with moving sensors and actuators. Our proposed algorithm has been implemented on `Diff-MAS2D`. Simulation result shows the effectiveness of our algorithm.

The remaining part of this paper is organized as follows: In Sec. II, the problem formulation is presented. In Sec. III, we give a brief introduction of Voronoi diagram and Centroidal Voronoi Tessellation. Some related properties of CVT are described. Section IV is devoted to introducing our simulation platform `Diff-`

Corresponding author: Prof. YangQuan Chen, Center for Self-Organizing and Intelligent Systems, Dept. of Electrical and Computer Engineering, 4160 Old Main Hill, Utah State University, Logan, UT 84322-4160. **T**: (435)7970148, **F**: (435)7973054, **E:** `yqchen@ece.usu.edu`; **W**: `www.csois.usu.edu`; `mechatronics.ece.usu.edu/mas-net/`

0-7803-9568-9/05/$20.00 ©2005 IEEE

2D for PDE system measurement and control with mobile sensors and mobile actuators. In Sec. V, we present the algorithms used in the simulation. To show the effectiveness of our proposed method, experimental results are presented in Sec. VI. Finally, conclusions and future research directions are presented in Sec. VII.

II. Problem formulation

In this section, the problem of robot optimal position and trajectory generation in feedback control for pollution neutralization is formulated.

Let Ω be a convex polytope in $\mathcal{R}^2$, including its interior. A concentration function is a map $\rho(x,y):\Omega\rightarrow\mathcal{R}_+$ that represents the pollutant concentration over Ω. To simplify the presentation of our main idea, in our simulation experiment, we assume $\rho(x,y)$ is governed by the following PDE system:

$$\frac{\partial\rho}{\partial t}=k\left(\frac{\partial^2\rho}{\partial x^2}+\frac{\partial^2\rho}{\partial y^2}\right)+f_d(\rho,x,y,t), \qquad (1)$$

where k is a constant positive real system parameter; $f_d(\rho,x,y,t)$ represents the source of the pollution. Currently, there is no control input applied to the system. We assume that the diffusion process is evolving slowly.

Let $P=(p_1,\cdots,p_n)$ be the location of n actuators and let $|\cdot|$ denote the Euclidean distance function. Every robot at p_i will receive information of sensors and release the neutralization chemical by some control law. The objectives are:

- Control the diffusion of the pollution to a confined area.
- Neutralize the pollution in a time optimal way while not making the area of interest overdosed.
- Minimize the polluted area that is more heavily affected.

n robots will partition Ω into a collection of n polytopes $\mathcal{V}=\{V_1,\cdots,V_n\}$, $p_i\in V_i$, $V_i\cap V_j=\emptyset$ for $i\neq j$ and $\cup_{i=1}^n\bar{V}_i=\bar{\Omega}$ ($\bar{V}_i=V_i\cup\partial V_i$ and $\bar{\Omega}=\Omega\cup\partial\Omega$). It can be seen that to control the diffusion process and minimize the heavily affected area, the robots should be close to those areas with high pollution concentrations so that the pollution can be neutralized timely and does not diffuse further. They can be far from the lightly polluted areas. But putting all robots very close to the pollution source is not a good strategy, because the diffused pollutants that are far away from the source can not be neutralized timely. To decide the positions of the robots, we consider the minimizing of the following cost function

$$\mathcal{K}(P,\mathcal{V})=\sum_{i=1}^{n}\int_{V_i}\rho(q)|q-p_i|^2\mathrm{d}q \text{ for } q\in\Omega. \qquad (2)$$

It is clear that to minimize $\mathcal{K}$, the distance $|q-p_i|$ should be small when the pollution concentration $\rho(q)$ is big. It is the concentration function $\rho(q)$ that determines the optimal positions of the robots. A necessary condition for $\mathcal{K}$ to be minimized is that $\{p_i,V_i\}_{i=1}^k$ is a Centroidal Voronoi Tessellation of Ω [14]. Our algorithm is based on a discrete version of (2) and the concentration information comes from the measurements of the static, low-cost sensors.

III. Voronoi Diagram and Centroidal Voronoi Tessellation

Here we give a brief introduction to the Voronoi diagram and Centroidal Voronoi Tessellation [14].

Given an open set $\Omega\subset\mathcal{R}^N$ and a set of points $\{z_i\}_{i=1}^k$ belonging to $\bar{\Omega}$, let $|\cdot|$ denote the Euclidean norm in $\mathcal{R}^N$ and let

$$V_i=\{x\in\Omega||x-z_i|<|x-z_j| \text{ for } j=1,\cdots,k,j\neq i\} \qquad (3)$$
$$i=1,\cdots,k.$$

It is easy to see that

$$V_i\cap V_j=\emptyset \text{ for } i\neq j \text{ and } \cup_{i=1}^n\bar{V}_i=\bar{\Omega}.$$

The set $\{V_i\}_{i=1}^k$ is referred to as a Voronoi tessellation or Voronoi diagram of Ω and each V_i is referred to as the Voronoi region or Voronoi cell. The members of the set $\{z_i\}_{i=1}^k$ are referred to as generators of each cell V_i, as shown in Fig. 1(a).

Given a density function $\rho(x)\geq 0$ defined on $\bar{\Omega}$, then for each Voronoi cell V_i, we define the mass centroid z_i^* of V_i by:

$$z_i^*=\frac{\int_{V_i}x\rho(x)\mathrm{d}x}{\int_{V_i}\rho(x)\mathrm{d}x} \text{ for } i=1,\cdots,k. \qquad (4)$$

We call the tessellation defined by (3) a Centroidal Voronoi Tessellation if and only if

$$z_i=z_i^* \text{ for } i=1,\cdots,k.$$

So, the points z_i that serves as the generators for the Voronoi regions V_i are themselves the mass centroids of those regions, as shown in Fig. 1(b).

Centroidal Voronoi Tessellation has broad applications in many fields. It is the solution to optimal placement of resources, but in general, CVT can only be approximately constructed. For algorithms to implement CVT, refer to [14]. In [12], an example is given to show how CVT can be used to predict the cell divisions. It is shown that, after the cell division process, the new cells' shapes are very closely approximated by Centroidal Voronoi Tessellations corresponding to the increased number of generators. This is an example to show how CVT can be applied in a dynamically evolving environment.

IV. Simulation Platform Diff-MAS2D [13], [15]

In this section, we introduce the simulation platform, Diff-MAS2D [13], [15] for simulation of measurement scheduling and controls in distributed parameter systems with moving sensors and moving actuators that other mathematical tools, for example, MATLAB PDE Toolbox [16], FEMLAB [17], Nastran [18], ANSYS [19] can not solve easily.

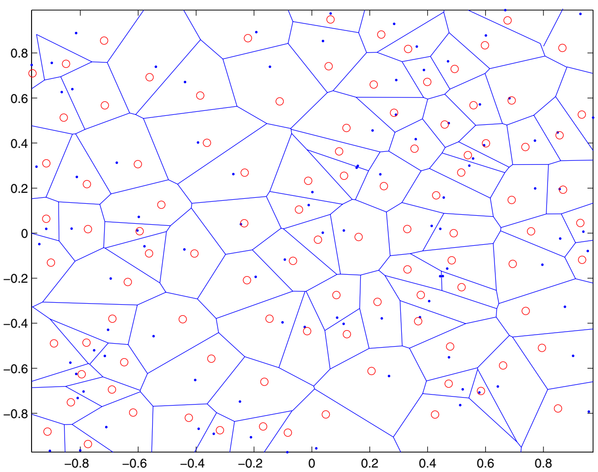
Specifically, Diff-MAS2D is used to solve the following parabolic PDE:

$$\frac{\partial\rho}{\partial t}=k\left(\frac{\partial^2\rho}{\partial x^2}+\frac{\partial^2\rho}{\partial y^2}\right)+f(\tilde{\rho},x,y,t), \qquad (5)$$

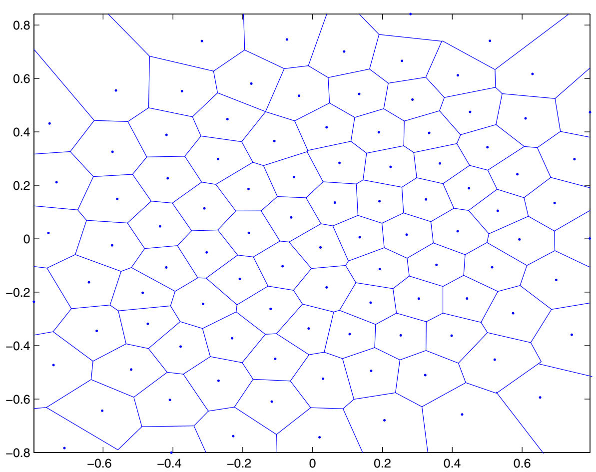
where $\rho=\rho(x,y,t)$ is the variable to be controlled; $0\leq x\leq 1$ and $0\leq y\leq 1$ is the spatial domain; $t\geq 0$ is the time domain; k is a positive real constant system parameter; $f(\tilde{\rho},x,y,t)$ is a combination of control and disturbances (pollution source).

$$f(\tilde{\rho},x,y,t)=f_c(\tilde{\rho}(x,y,t),x,y,t)+f_d(x,y,t),$$

where $\tilde{\rho}(x,y,t)$ is the measured data of $\rho(x,y,t)$ from the movable sensors; $f_c(\tilde{\rho}(x,y,t),x,y,t)$ is the control applied by the movable actuators; $f_d(x,y,t)$ is the disturbance to model the pollution source. The exact format of $f_c(\tilde{\rho}(x,y,t),x,y,t)$ depends on the closed-loop control law designed by the user based on certain control performance requirement.



(a) The Voronoi regions constructed by 100 randomly selected points in a square $\Omega = (-1, 1)^2$. The density function is given by $\rho(x, y) = e^{-8(x^2+y^2)}$. The dots are the Voronoi generators and the circles are the mass centroids of the corresponding Voronoi regions. The generators and the mass centroids do not coincide.



(b) The centroidal Voronoi tessellations constructed by 100 points in the same Ω and same density function ρ as in 1(a). These points server both as the generators and the mass centroids of the corresponding regions. More of them aggregate to area where $\rho(x, y)$ is bigger.

Fig. 1. The illustration of Voronoi diagram and Centroidal Voronoi Tessellations.

Arbitrary combination of the following two types of boundary conditions can be used as boundary conditions for each boundary ($x = 0$, $x = 1$, $y = 0$, and $y = 1$).

- Dirichlet boundary condition

$$\rho = C \tag{6}$$

  where C is a real constant.
- Neumann boundary condition

$$\frac{\partial \rho}{\partial n} = C_1 + C_2 \rho \tag{7}$$

  where C_1 and C_2 are two real constants; n is the outward direction normal to the boundary.

`Diff-MAS2D` uses finite-difference method to discretize the spatial domain of the diffusion equation and leaves the time domain integration to Matlab/Simulink. This scheme enables both `Diff-MAS2D` designers and end-users to make fully use of the capabilities of Matlab/Simulink. The whole function set of Matlab and Matlab Toolbox is available to the end users in designing sensor/actuator trajectories and designing the control laws of the actuators to control the diffusion process. The main features of `Diff-MAS2D` are listed as follows.

- Sensors and actuators can be collocated (bound together) or non-collocated (separated).
- Disturbances can be movable and time-varying.
- The mobility platform dynamics of sensors and actuators can be modelled as either first order (single integrator) or second order (double integrator).
- Movement of sensors and actuators can be open-loop (designed by the user as functions of time only) or closed-loop (designed by the user as functions of t, $\tilde{\rho}(x, y, t)$, sensor position/velocity, and actuator position/velocity).
- Arbitrary control algorithms can be applied in $f_c(\tilde{\rho}(x, y, t), x, y, t)$.

V. Proposed Optimal Actuator Locations Algorithms

First, we describe the algorithms to compute the locations of robots by Centroidal Voronoi Tessellations. Lloyd's method is a deterministic algorithm for determining Centroidal Voronoi Tessellations and is described below [14]:

Given a region Ω, a density function $\rho(x)$ defined for all $x \in \bar{\Omega}$, and a positive integer k

1) Select an initial set of k points $\{z_i\}_{i=1}^k$ as the generators.
2) Construct the Voronoi sets $\{V_i\}_{i=1}^k$ associated with generators $\{z_i\}_{i=1}^k$;
3) Determine the mass centroids of the Voronoi sets $\{V_i\}_{i=1}^k$; these centroids form the new set of points $\{z_i\}_{i=1}^k$;
4) If the new points meet some convergence criterion, terminate; other wise, return to step 2.

Although the CVT is used to solve the static resource location problem, if the diffusion process evolves slowly compared with the convergence rate of the Lloyd's method and the control efforts, CVT is still a valid solution to our problem, as verified in our simulation results presented in Sec. VI. The Lloyd's method will be executed periodically so that the motion of the robots can be adaptive to the evolution of the diffusion process.

Lloyd's method converges fast compared with probabilistic methods, for example, the MacQueen's method [14]. It requires substantially fewer iterations, but it has higher computation requirements for each iteration. In many applications, the robot has only limited communication abilities. To avoid communication collision and reduce computation requirement, a distributed asynchronous algorithm to construct the Voronoi diagram based on local information is clearly more desirable. We assume that the robot can communicate with the sensors and other robots within radius R_i. R_i is an adjustable parameter. Here we introduce a distributed algorithm from [6] with mild modification. At the first execution of step 2 in the above Lloyd's algorithm, each robot will do the following:

1) Assign its detection range R_i with a small initial value, detect all its neighboring robots with radius R_i.
2) Construct its own Voronoi cell within the radius R_i.
3) For every sensor q_i, compute $d_i = \max|q_i - p_i|$.
4) If $R_i > 2 \times d_i$, stop. Otherwise set $R_i = 2 \times R_i$, go to step 2.

R_i obtained at the first execution will be used as the initial values for the following executions. If for some time, for example, successive 3 updates, the R_i remains unchanged, then R_i can be decreased, $R_i = R_i - \Delta r$ for some $\Delta r > 0$. This improvement on the algorithm from [6] helps to reduce the computation requirements.

The mobile robots are treated as virtual particles and obey the second-order dynamical equation:

$$\ddot{p}_i = F_i$$

where F_i is given by

$$F_i = f_i - k_v \dot{p}_i \tag{8}$$

with f_i a force input to control the motion of the robot given by the following proportional control law:

$$f_i = -k(p_i - \bar{p}_i)$$

where $\bar{p}_i$ is the computed mass centroid of the current Voronoi cell.

The second term of (8) on the right hand side is the viscous friction artificially introduced [10]. k_v is the friction coefficient and $\dot{p}_i$ denotes the velocity of the robot i. This term is used to eliminate the oscillatory behavior of robots described in [7] when the robot is close to its destination. The viscous term assures that in the absence of the external force, the robot will come to a standstill state eventually.

We can also use proportional control for the neutralizing chemical releasing. The amount of chemicals each robot releases is proportional to the average pollutant concentration in the Voronoi cell belonging to that robot. Although our simulation is model-based, our control algorithms for each robot are not relying on the exact model information. They are based only on the sensor information that the robots can access.

VI. Simulation Results

`Diff-MAS2D` is used as the simulation platform for our implementation. The area concerned is given by $\Omega = \{(x, y)|0 \leq x \leq 1, 0 \leq y \leq 1\}$.

The system with control input is modelled as

$$\frac{\partial \rho(x,y,t)}{\partial t} = k\left(\frac{\partial^2 \rho(x,y,t)}{\partial x^2} + \frac{\partial^2 \rho(x,y,t)}{\partial y^2}\right) + f_c(x,y,t) + f_d(x,y,t), \tag{9}$$

where $k = 0.01$ and the boundary condition is given by

$$\frac{\partial u}{\partial n} = 0.$$

The stationary pollution source is modelled as a point disturbance f_d to the the PDE system (9) with its position at $(0.75, 0.35)$ and

$$f_d(t) = 20e^{-t}|_{(x=0.75,y=0.35)}.$$

In our simulation, we assume that once deployed, the sensors remain static. There are 29×29 sensors evenly distributed in a square area $(0, 1)^2$ and they form a mesh over the area. There are 4 robots that can release the neutralizing chemicals. For the robot motion control, the viscous coefficient is given by $k_v = 1$ and the control input is given by

$$F_i = -3(p_i - \bar{p}_i) - \dot{p}_i.$$

The pollution source begins to diffuse at $t = 0$ to the area Ω, 4 robots are deployed with initial positions at $(0.33, 0.33), (0.33, 0.66), (0.66, 0.33), (0.66, 0.66)$, respectively. Figure 2 shows the initial positions of the robots, the positions of the sensors and the position of the pollution source.

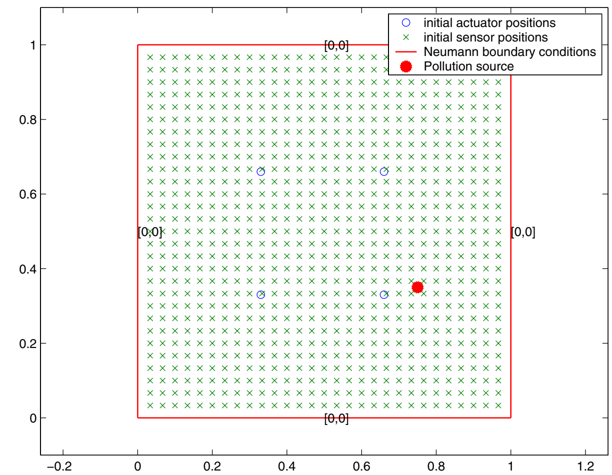


Fig. 2. Initial layout of actuators and sensors.

We choose the simulation time to $t = 5$s and the time step is chosen as $\Delta t = 0.002$s. The robot recomputes its desired position every 0.2s. To show how the robots can control the diffusion of the pollutants, the robots begin to react at $t = 0.4$s. The system evolves under the effects of diffusion of pollutants and diffusion of neutralizing chemicals released by robots. In Fig. 3, the y axis is the sum of the sensor measurements. It shows that the amount of pollutants decreases to 28% of its peak value at the end of the simulation. And the the decreasing process is monotonic. The evolution of the amount of pollutants without control is also shown in Fig. 3.

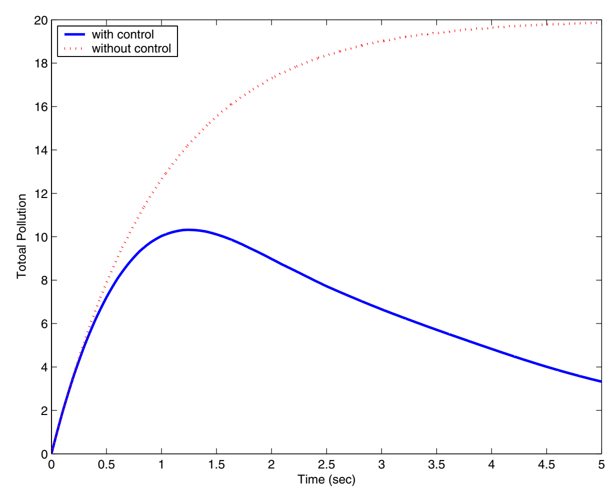


Fig. 3. Evolution of the amount of pollutants (with and without control).

Figure 4 shows the trajectories of the robots for $t \leq 5$s. It can be seen that the robots move towards the pollution source to suppress the diffusion of the source. And then they move around to track the pollution that has already diffused and try to neutralize it.

Figure 5 shows the evolution of the diffusion process at $t = 1s, 1.5s, 2.0s, 2.5s, 3.0s, 3.5s, 4.0s, 4.5s$, respectively. The left part of the figure is a 3D plot of $\rho(x, y, t)$ and the right part

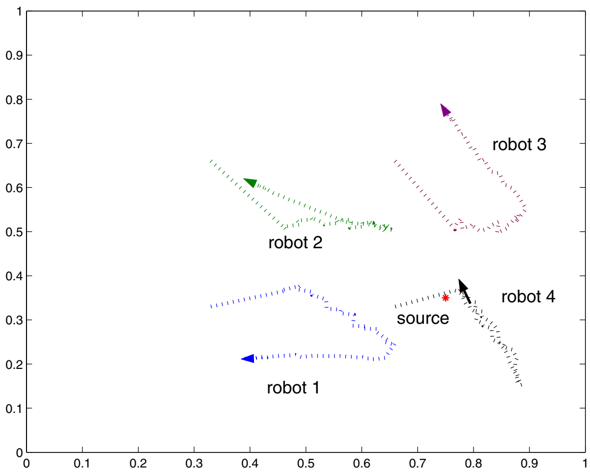
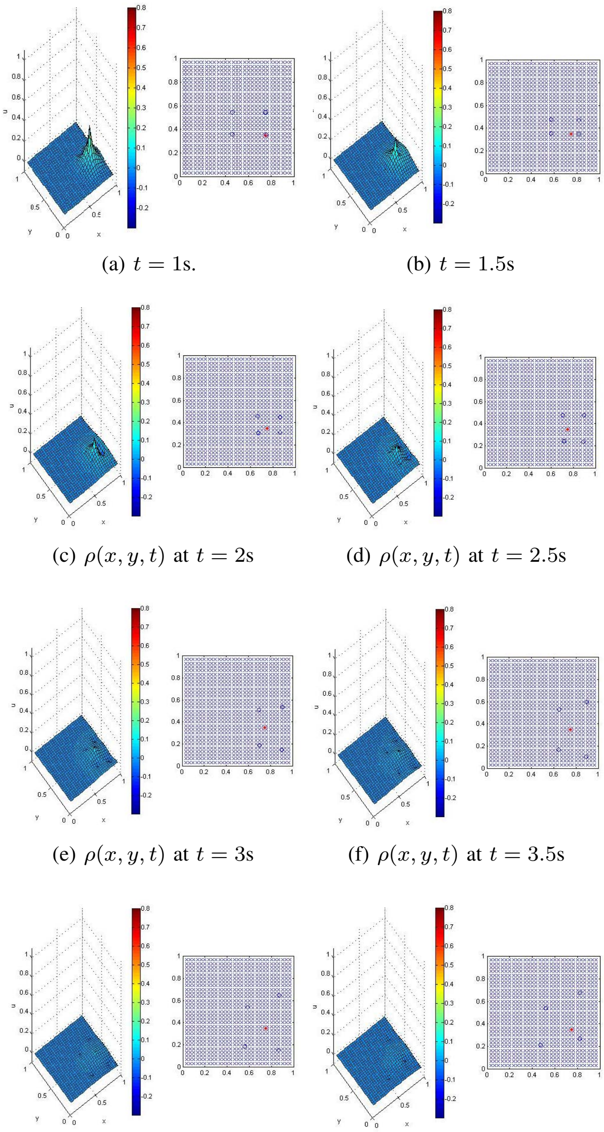


Fig. 4. Robots trajectories



(a) $t = 1$s. (b) $t = 1.5$s

(c) $\rho(x, y, t)$ at $t = 2$s (d) $\rho(x, y, t)$ at $t = 2.5$s

(e) $\rho(x, y, t)$ at $t = 3$s (f) $\rho(x, y, t)$ at $t = 3.5$s

(g) $\rho(x, y, t)$ at $t = 4$s (h) $\rho(x, y, t)$ at $t = 4.5$s

Fig. 5. Evolution of $\rho(x, y, t)$ and mobile robots positions

of the figure shows the current position of the mobile robots. In Fig. 5(a), robot 4 moves very close to the pollution source. There is a peak at the pollution source. In Fig. 5(f), the peak is suppressed and the robots move towards the boundary of Ω.

Figure 6 shows the evolution of Voronoi diagrams at those time instants.

To show the effectiveness of our proposed algorithm for pollution neutralization, we compare our method based on CVT with the case when 4 robots are uniformly distributed at at $(0.33, 0.33), (0.33, 0.66), (0.66, 0.33), (0.66, 0.66)$ respectively and keep still. The control laws for chemical releasing are the same. We assume that if the concentration of the pollution ranges in $0.025 \geq \rho(x, y, t) \geq 0$, it is safe. In Fig. 7, the `y` axis represents the percentage of sensors whose measurements are at safe level. It can be seen that the safe area increases fast after the control has been implemented by method based on CVT. It means the polluted area can be recovered quickly. While by using evenly distributed deployment, the robots can not recover the polluted area efficiently. We need to point out that, with the control laws for robot motion and for chemical releasing, only isolated sensors report $\rho(x, y, t) \leq 0$ and the number is $< 2\%$ of the total number of sensors. So the assumption that $\rho(x, y, t)$ is ≥ 0 for mass centroids computation is kept valid.

VII. Conclusion

In this paper, we extend the application of Centroidal Voronoi Tessellation to pollution elimination by distributed feedback control with moving actuators and a set of static mesh sensors. A simulation platform `Diff-MAS2D` is used to illustrate the effectiveness of our algorithm. In the future, we will extend our research for pollution feedback control by using mobile sensors and mobile actuators together. We will take into account the sensor noise and unreliable communication and try to exploit a model-based approach.

References

[1] Kevin L. Moore, Yang Quan Chen, and Zhen Song, "Diffusion based path planning in mobile actuator-sensor networks (MAS-net) - some preliminary results," in *Proc. of SPIE Conf. on Intelligent Computing: Theory and Applications II, part of SPIE's Defense and Security*, Orlando, FL., USA, April 2004, SPIE.

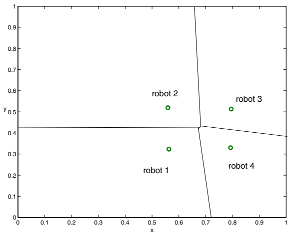
[2] YangQuan Chen, Kevin L. Moore, and Zhen Song, "Diffusion boundary determination and zone control via mobile actuator-sensor networks (MAS-net): Challenges and opportunities," in *Intelligent Computing: Theory and Applications II, part of SPIE's Defense and Security*, Orlando, FL, Apr. 2004.

[3] Zhongmin Wang, Zhen Song, Peng-Yu Chen, Anisha Arora, Dan Stormout, and YangQuan Chen, "MASmote - a mobility node for MAS-net (mobile actuator sensor networks)," in *Proceedings of 2004 IEEE International Conference on Robotics and Biomimetics*, Shengyang, China, August 22-25 2004, Robio04.

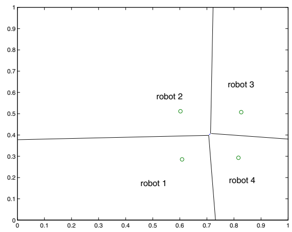
[4] K. Moore and Y. Chen, "Model-based approach to characterization of diffusion processes via distributed control of actuated sensor networks," in *IFAC Symposium of Telematics Applications in Automation and Robotics, Helsinki University of Technology Espoo, Finland*, 2004.

[5] Ioannis Rekleitis, Vincent Lee-Shue, Ai Peng New, and Howie Choset, "Limited communication, multi-robot team based coverage," in *Proceedings of the 2004 IEEE International Conference on Robotics and Automation*, New Orleans, LA, April 2004, pp. 3462–3468.

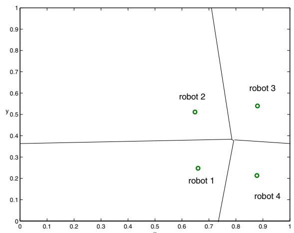
[6] Timur Karatas Jorge Cortés, Sonia Matínez and Francesco Bullo,



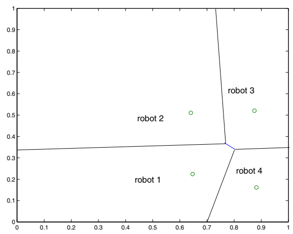
(a) Voronoi diagram at $t = 1$s.



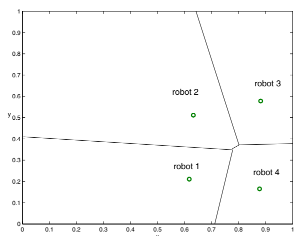
(b) Voronoi diagram at $t = 1.5$s



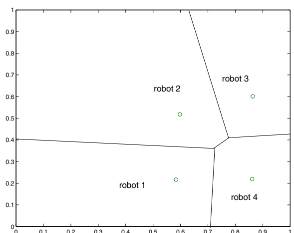
(c) Voronoi diagram at $t = 2$s



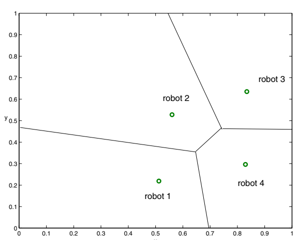
(d) Voronoi diagram at $t = 2.5$s



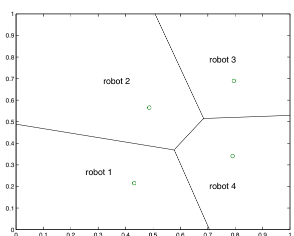
(e) Voronoi diagram at $t = 3$s



(f) Voronoi diagram at $t = 3.5$s



(g) Voronoi diagram at $t = 4$s



(h) Voronoi diagram at $t = 4.5$s

Fig. 6. Evolution of Voronoi diagrams

"Coverage Control for Mobile Sensing Networks," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 20, pp. 243–255, April 2004.

[7] Nojeong Heo and P. K. Varshney, "Energy-efficient deployment of intelligent mobile sensor networks," *IEEE Transactions on Man and CyberneticsSystems*, vol. Part A, pp. 78 – 92, January 2005.

[8] Jeanne A. Atwell and Belinda B. King, "Reduced order controllers for spatially distributed systems via proper orthogonal decomposition," *SIAM Journal on Scientific Computing*, vol. 26, no. 1, pp. 128–151, 2004.

[9] A. L. Faulds and B. B. King, "Sensor location for feedback control of partial differential equation systems," in *Proceedings of the 2000 IEEE CCA/CACSD*, Anchorage, AK, September 2000, pp. 536 – 541.

[10] Andrew Howard, Maja J Mataric, and Gaurav S Sukhatme, "Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem," in *Proceedings of the 6th International Symposium on Distributed Autonomous Robotics Systems*, Fukuoka, Japan, June 2002.

[11] Timothy H. Chung, Vijay Gupta, Joel W. Burdick, and Richard M. Murray, "On a decentralized active sensing strategy using mobile

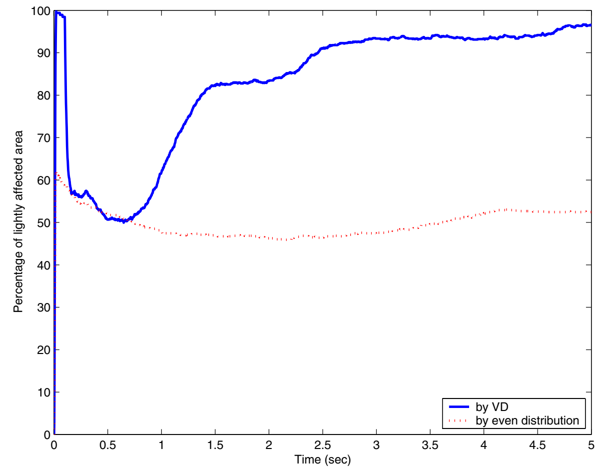


Fig. 7. Comparison of two different robot deployment methods.

sensor platforms in a network," in *Proceedings of 43th IEEE Conference on Decision and Control*, December 2004, pp. 1914–1919.

[12] Qiang Du, Vance Faber, and Max Gunzburger, "Centroidal Voronoi Tessellations: Applications and Algorithms," *SIAM REVIEW*, vol. 41, no. 4, pp. 637–676, 1999.

[13] Jingsong Liang and YangQuan Chen, "Diff-MAS2D (version 0.9) user's manual: A simulation platform for controlling distributed parameter systems (diffusion) with networked movable actuators and sensors (MAS) in 2D domain," Tech. Rep. USU-CSOIS-TR-04-03, CSOIS, Utah State University, 2004.

[14] Lili Ju, Qiang Du, and Max Gunzburger, "Probabilistic methods for centroidal voronoi tessellations and their parallel implementations," *Parallel Comput.*, vol. 28, no. 10, pp. 1477–1500, 2002.

[15] Jinsong Liang and YangQuan Chen, "`Diff/Wave-MAS2D`: A simulation platform for measurement and actuation scheduling in distributed parameter systems with mobile actuators and sensors," in *Proceedings of the 2005 IEEE Int. Conf. on Mechatronics and Automation (ICMA)*, Niagara Falls, Canada, July 2005.

[16] "MATLAB," `http://www.mathworks.com/`.

[17] "Femlab," `http://www.comsol.com/`.

[18] "Nastran," `http://www.nenastran.com`.

[19] "ANSYS," `http://www.ansys.com`.