Fall 2016

Using wireless sensors and networks program for chemical particle propagation mapping and chemical source localization

Xiang Gao

Department: Electrical and Computer Engineering

USING WIRELESS SENSORS AND NETWORKS PROGRAM FOR CHEMICAL PARTICLE PROPAGATION MAPPING AND CHEMICAL SOURCE LOCALIZATION

by

XIANG GAO

A DISSERTATION

Presented to the Faculty of the Graduate School of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

DOCTOR OF PHILOSOPHY

in

ELECTRICAL ENGINEERING

2016

Approved
Levent Acar, Advisor
Jagannathan Sarangapani, Co-advisor
Maciej J. Zawodniok
Zhaozheng Yin
Jie Huang

ABSTRACT

Chemical source localization is a challenge for most of researchers. It has extensive applications, such as anti-terrorist military, Gas and oil industry, and environment engineering. This dissertation used wireless sensor and sensor networks to get chemical particle propagation mapping and chemical source localization. First, the chemical particle propagation mapping is built using interpolation and extrapolation methods. The interpolation method get the chemical particle path through the sensors, and the extrapolation method get the chemical particle beyond the sensors. Both of them compose of the mapping in the whole considered area. Second, the algorithm of sensor fusion is proposed. It smooths the chemical particle paths through integration of more sensors' value and updating the parameters. The updated parameters are associated with including sensor fusion among chemical sensors and wind sensors at same positions and sensor fusion among sensors at different positions. This algorithm improves the accuracy and efficiency of chemical particle mapping. Last, the reasoning system is implemented aiming to detect the chemical source in the considered region where the chemical particle propagation mapping has been finished. This control scheme dynamically analyzes the data from the sensors and guide us to find the goal. In this dissertation, the novel algorithm of modelling chemical propagation is programmed using Matlab. Comparing the results from computational fluid dynamics (CFD) software COMSOL, this algorithm have the same level of accuracy. However, it saves more computational times and memories. The simulation and experiment of detecting chemical source in an indoor environment and outdoor environment are finished in this dissertation.

ACKNOWLEDGMENTS

Five and half year's research is not a short time. During my Ph.D studying, my advisor Dr. Levent Acar gave me a lot of help and guide. Without his hard work, I couldn't finish my research. The knowledge and research methods that he taught me are the precious treasure for me in the future life. That means a lot for me.

My co-advisor Dr. Sarangapani, Jagannathan taught me professional knowledges and technical writing skill that were used in my research.

I want to thank my committee members, Dr. Maciej J Zawodniok, Dr. Zhaozheng Yin and Dr. Jie Huangthey gave me a lot of good advices that helps my dissertation.

Finally, thanking my father Zengsuo Gao and mother Meiling Zhang, they gave me unconditional support for my research and life.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

1. INTRODUCTION

1.1. STATE OF THE RELATED WORK

There are a lot of previous works that have been done in this field about that using wireless sensors and networks program for chemical source localization. Plenty of papers are talking about smell propagation, smell detection, smell localization, smell modelling.

1.1.1. Technologies about Smell Propagate, Smell Transmit, Smell Spread. In the paper [1], Virtual slaughtering and dissection is a way to learn about anatomy and dissect animals without harming any real-life animals or feeling ill from that nauseating formaldehyde smell. The virtual slaughtering and dissections also have detailed scientific explanations. The authors developed interactive virtual environments that can simulate several common tasks performed during animal dissection. In this paper, it describes the imaging modality used to reconstruct the cow, provide an overview of the simulation environment and briefly discuss some of the techniques used to manipulate the virtual cow. The system will consists of an anatomically computer model of a cow, a simulation engine capable of providing soft-tissue modeling, rigid body dynamics, collision detection and response, haptic force calculations, and a number of user interface and display devices to interact with the user. The behavior of the tissue will be modeled by modulating the stiffness coefficients between adjacent internal organs in order to propagate the effects of grasping connected components. This simulation system will be written in C# using Open GL, GLUT and GLUI libraries for visualization and user interface. Several supporting virtual reality tools for grasping, cutting and probing will be used such as haptic devices, non-haptic devices, and display devices.

In this paper [2], digital image processing techniques to detect smell and PH value as well as surface color quality of lamb and other information was used. The article analyses and establishes multiple data fusion model with artificial neural network training. And designs finally corresponds to the lamb freshness standard TVB-N. Rapid and non-destructive testing of lamb freshness was researched.

In the paper [3], a low cost smell detection system for multimedia applications are proposed. A multimedia system is one of the most widely used consumer electronics

environments today in which humans can work or communicate through multi-sensory interfaces. Existing multimedia systems include audio systems and video displays that make use of the human auditory and visual systems. The minor sensory information namely, smell, taste and touch are generally not used. In order to use smell in multimedia environments, smell detectors and producers are required. The author propose a low cost smell detector that can be used in a multimedia environment. It is shown that a crude smell detection system, which can detect tens of smells can be developed at a low cost, compared to that of typical systems used in industrial applications.

In the paper [4], the author discussed Odor Recognition for Intelligent Systems. Smell, it helps remind us when it's time to take out the garbage or change the baby's diaper. The author use it to detect danger, such as a gas leak or food burning on the stove. Industries that develop products to help us either smell good or prevent aromatic offense testify to the aesthetic importance of odors, as well as to economic value. Still, olfaction's significance is unparalleled in the animal kingdom, where many species' survival would be jeopardized without the ability to detect and recognize odors. Integrating electronic noses with other sensors on complex, intelligent platforms offers exciting application possibilities and considerable development challenges.

In the paper [5], a guide to sensor design for land mine detection is discussed. The author discusses the use of smell by dogs to detect land mines. The author then discusses the concept of using a man-made odor detector to detect explosive vapors. The author highlights one sensor type which shows outstanding potential for land mine detector use at present: this is the conducting polymer sensor. This type of sensor has a high degree of sensitivity and other desirable features and has been in operation for some years now by a British company for the purpose of aroma analysis and quality control of industrial and agricultural products. It would certainly be possible to develop this type of sensor further for land mine detection. The principle of biological smell detection is outlined.

Early stage fire detection using reliable metal oxide gas sensors and artificial neural networks is addressed in the paper [6]. Conventional fire detectors use the smoke density or the high air temperature to trigger the fire alarm. These devices lack of ability to detect the source of fire in the early stage and they always create false alarms. In this paper, a reliable electronic nose (EN) system designed from the combination of various

metal oxide gas sensors (MOGS) is applied to detect the early stage of fire from various sources. The time series signals of the same source of fire in every repetition data are highly correlated and each source of fire has a unique pattern of time series data. Therefore, the error backpropagation (BP) method can classify the tested smell with 99.6% of correct classification by using only a single training data from each source of fire. The results of the k-means algorithms can be achieved 98.3% of correct classification which also show the high ability of the EN to detect the early stage of fire from various sources accurately.

Instant coffee classification by electronic noses has been implemented in the paper [7]. Normally, an electronic nose project uses two researches areas which are hardware for developing sensors to detect substance smell and software using pattern matching theorem for recognizing substance. For this research, the operation begins with sensors hit the coffee smell. The result is converted from analog to digital representation. An artificial intelligence is a tool of a thinking system which can create knowledge as if a human does. The objective of this research is to classify instant coffee by using electronic noses. the author used eight types of coffee in Thailand market for this project which are (1) Moccona-select, (2) Moccona-royal gold, (3) Nescafe redcup, (4) Nescafe gold, (5) Khao Shong brown, (6) Khao Shong red, (7) Oem-Big C and (8) Superclass. The author compared four structures of neural network to classify the coffee data. The precision of correctness is equal to 65.63 for a neural network structure as 7 input-layer nodes, 14 hidden-layer1 nodes, 48 hidden-layer2 nodes and 8 output-layer nodes.

Micro and Nano sensors snoop around are talked in the paper [8]. Vision and hearing, smell and taste, and the tactile senses are bridges between the external world and our brain. Micro and Nano sensors are miniaturized electronic devices which pick up physical, chemical, or biomedical signals and enter them into the computer. The miniaturization of most kinds of sensors has been achieved, but the "electronic nose" able to detect a broad range of "smells" caused by complex mixtures of airborne chemical compounds is still a dream. But application specific gas sensors or "narrow band noses" are being developed, which can detect and identify gas mixtures in given application areas, such as air conditioning, dry cleaning, oil refineries, or food production. Integrated gas sensors based on CMOS IC technology with on-chip micro structures (CMOS

MEMS) coated with gas absorbing polymers or metal oxides are presented. The quest of sensor selectivity is tackled by combining various polymers with different transducer principles (mass sensitive, capacitive, calorimetric). The combination of different types of transducers on a single CMOS MEMS chip with dedicated circuitry, and the assembly of several such chips, each with different chemically sensitive polymer layer, in a handheld "snooping instrument" are discussed. An outlook will address the combination of CMOS MEMS with bio materials and living cells.

In the paper [9] ‘Smell Peak Prediction during Black Tea Fermentation Process Using Time-Delay Neural Network on Electronic Nose Data’, Fermentation process in black tea manufacturing plays the key role in determining the quality of finished tea. During this process, a complex chain of biochemical changes occurs and the process should be terminated once the optimum fermentation point is reached. Present day practice for detection of optimum fermentation point is purely subjective, and is carried out by experienced industry personnel. Even though chemical methods are available, but they are expensive, time-consuming and offline. A study has been made on real time smell monitoring of black tea during fermentation process using electronic nose and is reported in this paper. Time-delay neural network (TDNN) architecture has been used on time series data obtained from electronic nose for smell peak prediction during the fermentation process. The online predicted result using TDNN seems very promising to detect the optimum fermentation time for black tea manufacturing process.

1.1.2. Technologies about Smell Detection. Gas sensors provide an artificial sense of smell for a mobile robot to track an airborne gas/odor plume and locate its source. However, a slow response of gas sensors has been the major factor limiting the development of plume-tracking robots. This paper [10] describes a new control algorithm that breaks the limitation. The basic idea is to detect onsets of gas sensor response and starts of recovery by monitoring the relative change in each sensor output. Fast plume tracking is accomplished by making the robot take appropriate actions immediately when the sensor outputs start changing from one state to another. Growing sensor outputs evoke an increase in the robot speed for further acceleration of plume tracking, whereas insufficient sensor outputs slow down the robot to avoid degrading the search success rate. In contrast to the previous algorithm, based on the absolute sensor output levels, the

detection of output change also leads to reliable plume detection, since it is insusceptible to drift in the gas sensor outputs. Experimental results have shown that the robot can track down a gas source within the distance of 2 m in 32 s, even though semiconductor gas sensors with a long recovery time (>60 s) are used.

Multimedia systems are widely used in consumer electronics environments today, where humans can work and communicate through multi-sensory interfaces. Unfortunately smell detection and generation systems are not part of today's multimedia systems. In this paper [11], the author propose an electronic nose that can be used in a multimedia environment. The proposed electronic nose can detect a large number of smells and will have a significantly lower cost compared to the detection systems used in industrial applications.

A Bayesian Approach for the Detection of Code and Design Smells is presented in paper [12].The presence of code and design smells can have a severe impact on the quality of a program. Consequently, their detection and correction have drawn the attention of both researchers and practitioners who have proposed various approaches to detect code and design smells in programs. However, none of these approaches handle the inherent uncertainty of the detection process. The author propose a Bayesian approach to manage this uncertainty. First, the author present a systematic process to convert existing state-of-the-art detection rules into a probabilistic model. The author illustrate this process by generating a model to detect occurrences of the Blob antipattern. Second, the author present results of the validation of the model: the author built this model on two open-source programs, Gantt Project v1.10.2 and Xerces v2.7.0, and measured its accuracy. Third, the author compare our model with another approach to show that it returns the same candidate classes while ordering them to minimize the quality analysts' effort. Finally, the author show that when past detection results are available, our model can be calibrated using machine learning techniques to offer an improved, context-specific detection.

In the paper [13] Artificial Odor Discrimination System Using Electronic Nose and Neural Networks for the Identification of Urinary Tract Infection, Current clinical diagnostics are based on biochemical, immunological, or microbiological methods. However, these methods are operator dependent, time-consuming, expensive, and require

special skills, and are therefore, not suitable for point-of-care testing. Recent developments in gas-sensing technology and pattern recognition methods make electronic nose technology an interesting alternative for medical point-of-care devices. An electronic nose has been used to detect urinary tract infection from 45 suspected cases that were sent for analysis in a U.K. Public Health Registry. These samples were analyzed by incubation in a volatile generation test tube system for 4-5 h. Two issues are being addressed, including the implementation of an advanced neural network, based on a modified expectation maximization scheme that incorporates a dynamic structure methodology and the concept of a fusion of multiple classifiers dedicated to specific feature parameters. This study has shown the potential for early detection of microbial contaminants in urine samples using electronic nose technology.

An Immune-Inspired Approach for the Detection of Software Design Smells is been implemented in [14]. The author proposed a parallel between object-oriented system designs and living creatures. The author suggest that, like any living creature, system designs are subject to diseases, which are design smells (code smells and anti patterns). Design smells are conjectured in the literature to impact the quality and life of systems and, therefore, their detection has drawn the attention of both researchers and practitioners with various approaches. With our parallel, the author propose a novel approach built on models of the immune system responses to pathogenic material. The author show that our approach can detect more than one smell at a time. The author build and test our approach on Gantt Project v1.10.2 and Xerces v2.7.0, for which manually-validated and publicly available smells exist. The results show a significant improvement in detection time, precision, and recall, in comparison to the state-of-the-art approaches.

1.1.3. Technologies about Smell Localization and Smell Detection. Odor plume tracking robot using semiconductor gas sensors has been implemented [15]. This paper reports on the development of a mobile robot equipped with semiconductor gas sensors which has been designed to track odor plumes in the natural environment to locate their sources. One principal application could be the automatic location of explosives and analogies can be made with animal behavior, particularly dogs, which can track both objects and persons using smell. A novel multi-sensor head is proposed and the design of the mobile robot and its tracking strategies presented. The system is

evaluated using isopropyl alcohol as the odor source. Results are presented for a number of different environmental conditions.

To naively smell as no robot has smelt before. This paper [16] presents a new intelligent odor localization strategy, which enables a robot to locate the source of an odor in a cluttered indoor environment. Traditionally, work in this area has focused on open areas free of obstacles and having no walls or possessing walls without openings. Existing solutions predominantly use reactive algorithms to navigate along the entire length of the odor plume to the source. Not only is this slow, but in a cluttered indoor environment it may not be possible. In a constrained environment, airflows tend to circulate in sectors and well-defined plumes that lead upwind to the odor source do not exist. the author have developed a sense-map-plan-act style control strategy to model the airflow in the environment using naive physics, then use the model to reason about odor dispersal, move to key positions gathering information, and make a prediction of the most likely location for an odor source. The control strategy has located the odor source for a variety of room configurations. This paper describes details of the control strategy, practical experiments, and results.

Reproduction of scent and video at remote site using odor sensing system and olfactory display together with camera is a novel technology [17]. The author proposed the new system for scent reproduction at remote site together with video. The odor is identified using odor sensing system and the result is transferred via Internet to the remote site, where the identified smell is regenerated using an olfactory display. It is also possible to reproduce the odor concentration change. Users can sniff the smell in real time at the location away from the odor source, simultaneously watching a video around the odor source. The author did the experiment on the tele olfaction between our campus and Tokyo Big Sights. The questionnaire survey revealed that our system provided the good performance of the tele olfaction. The proposed system can enhance the reality when smell is attached to the video.

In the paper ‘Product metrics for automatic identification of "bad smell" design problems in Java source-code’ [18], Refactoring can have a direct influence on reducing the cost of software maintenance through changing the internal structure of the source-code to improve the overall design that helps the present and future programmers evolve

and understand a system. Bad smells are a set of design problems with refactoring identified as a solution. Locating these bad smells has been described as more a human intuition than an exact science. This paper addresses the issue of identifying the characteristics of a bad smell through the use of a set of software metrics. Then by using a pre-defined set of interpretation rules to interpret the software metric results applied to Java source-code, the software engineer can be provided with significant guidance as to the location of bad smells. These issues are addressed in a number of ways. Firstly, a precise definition of bad smells is given from the informal descriptions given by the originators Fowler and Beck. The characteristics of the bad smells have been used to define a set of measurements and interpretation rules for a subset of the bad smells. A prototype tool has been implemented to enable the evaluation of the interpretation rules in two case studies.

1.1.4. Technologies about Smell Modelling. An intelligent model approach for combination of sensor information has been built in the paper [19]. Humans gather information from the environment around them using different senses, e.g. sight, hearing, touch, smell and taste. By combining sensory information, the author are able to structure decisions and actions when interacting with the environment. Humans are capable of actively using perception capabilities in order to perform objectives in time and space. The objective of this paper is to discuss a biologically inspired sensor fusion model, named sensor fusion model with active perception (SEFMAP). The biological inspiration concept is not used to indicate biological plausibility in the sense of circuitry, networking architecture and information exchange modalities of proposed models, but the modeling point of view. SEFMAP has been developed by mimicking the human way of processing information received from sensory organs. This gives a simple and general model with great development potential, properties that in some degree are missing in existing models. SEFMAP was intended for modeling intelligent sensor fusion systems as well as traditional sensor fusion systems the model discussed in this paper, SEFMAP, includes three main processes (sensation, perception and active perception) as well as a knowledge base. SEFMAP reflects signal processing on sensory information that occurs on the way to the brain, as well as in the brain. The model also handles memory and decision-making to bring the system closer to an objective that may be changed during run-time. The

benefits of SEFMAP are demonstrated in three examples, a classification application, an auditory-visual target localization system and a fire indication system. The paper also considers how time affects the result of the sensor fusion algorithm.

Context-Aware Virtual Agents in Open Environments [20], this paper presents a model for the interaction between context-aware virtual agents and the environment in which they are situated. This model applies to multi-agent based simulation systems dealing with human-like virtual agents in decentralized, continuous, and dynamic environments. The model supports an extensible agent perception module, allowing agents to perceive their environment through multiple senses (sight, hearing, smell, etc.). The environment reacts to agent influences as well as user-invoked stimuli by combining these influences to determine the next state of the environment. This paper introduces a formalization and an implementation of the model and discusses multiple scenarios involving context-aware virtual agents situated in dynamic environments.

Signal Processing with temporal sequences in olfactory systems [21] is an import technology for this dissertation. The olfactory system is a very efficient biological setup capable of odor information processing with neural signals. The nature of neural signals restricts the information representation to multidimensional temporal sequences of spikes. The information is contained in the interspike intervals within each individual neural signal and interspike intervals between multiple signals. A mechanism of interactions between random excitations evoked by odorants in the olfactory receptors of the epithelium and deterministic operation of the olfactory bulb is proposed in this paper. Inverse Frobenius-Perron models of the bulb's temporal sequences are fitted to the interspike distributions of temporally modulated receptor signals. Ultimately, such pattern matching results in ability to recognize odors and offer a hypothetic model for signal processing occurring in the primary stage of the olfactory system.

‘Dynamical analysis of neural oscillators in an olfactory cortex model’, in this paper [22] presents a theoretical approach to understand the basic dynamics of a hierarchical and realistic computational model of the olfactory system proposed by W. J. Freeman. While the system's parameter space could be scanned to obtain the desired dynamical behavior, our approach exploits the hierarchical organization and focuses on understanding the simplest building block of this highly connected network. Based on

bifurcation analysis, the author obtain analytical solutions of how to control the qualitative behavior of a reduced KII set taking into consideration both the internal coupling coefficients and the external stimulus. This also provides useful insights for investigating higher level structures that are composed of the same basic structure. Experimental results are presented to verify our theoretical analysis.

An E-nose haar wavelet preprocessing circuit for spiking neural network classification has been built in paper [23]. A simulation model for polymer film chemical sensors is developed based on a 1 dimensional diffusion equation. Using this model, electronic nose smell prints produced by the 32 sensor array of a Cyranose 320 are simulated to test pattern classification. A Haar wavelet Alter reduces noise and captures information about the diffusion rate of the analyte in each sensor. Inputs are encoded into a binary Hamming pattern and fed into a binary spiking neural network for pattern classification. The preprocessing circuit for the spiking neural network, including the wavelet Alter, is designed using standard cells for a 180 nm process. Real and simulated results from the spiking neural network classification algorithm are favorably compared to Bayes, canonical, and PCA-PNN classifiers.

The visual based framework for the model refactoring techniques [24] is popular in today's research. Refactoring is one of the most important rules and practices of Extreme Programming from the family of the Agile Methodologies. The author propose the tool to refactor the UML model (Class Diagrams for now). In the first step the author need to find the flaws (bad smells) in the model with the OCL query and then in the second step the author transform the flaw to the correct fragment with the transformation script. The paper presents the set of methods and tools for the model adjustment, cooperating with the CASE systems. The author analyze the concept and algorithms for the refactoring, OCL queries and transformation scripts generating. The author have prepared functional prototype of the editor for the refactoring rules definition, OCL query generator and the transformation script generator. In the future, the author plan to extend the framework with alternative notations (e.g., QVT graph transformation rules, PICS, Viatra2) and the other techniques to find the flaws (e.g., rule-based system with predicates of the bad smells, XMI transformations and Abstract Syntax Tree algebra, Bit-Vector and Similarity Scoring Algorithms).

1.2. COMPARISON AMONG DIFFERENT TECHNOLOGIES FOR SMELL DETECTION AND LOCALIZATION

The problem of detecting and locating odor in a complicated environment is a challenging problem. The difficulty of modeling the propagation of the different odor sources and location of sensor has been addressed in many earlier works.

The ability to locate the source of an odor/chemical plume has many valuable applications. These applications include finding the source of dangerous substances such as airborne biological material, hazardous chemicals, gas and other pollutants, in industrial and other settings; detecting such things as plant matter and drugs in a customs or quarantine application; searching for survivors in earthquake-damaged buildings, landslides or avalanches; detecting fire in its initial stages; locating unexploded mines and bombs; and for inter-robot communication, particularly in robotic swarms.

While research into robotic odor localization only began in the early 90s, there are several research groups currently working in the area. Nowadays, there are two research groups who both have done some research about detecting and locating the odor source. The first group is from intelligent robotics research center, Monash University, Australia. The members are Gideon Kowadlo and R. Andrew Russell. The second group is from Department of Physical Electronics, Tokyo Institute of Technology, Japan. The members are Hiroshi Ishida, Gouki Nakayama, Takamichi Nakamoto, and Toyosaka Moriizumi. The following sections introduce two research groups' achievements and compare the two groups' differences.

The first group's members are Gideon Kowadlo and R. Andrew Russell. They have developed a sense-map-plan-act style control strategy to model the airflow in the environment using naïve physics, then use the model to reason about odor dispersal, move to key positions gathering information, and make a prediction of the most likely location for an odor source. The following describes details of airflow modeling, reasoning system for source prediction and experimental analysis.

A. Airflow modeling using naïve physics.

The airflow-modeling algorithm must be capable of producing a map of airflow that captures the broad features. It must divide the room into sectors of airflow including information of wind direction at the least, and if possible, also give information on the

velocity of the flow. The airflow modeling using naïve physics provide approximate information about patterns of airflow.

Naïve physics is essentially the use of common sense knowledge and physical intuition to model the environment, as opposed to the traditional approach of using differential equations. Generally speaking, this is a qualitative and not a quantitative approach.

They have developed a procedure for using such expressions, in our case in the form of naïve rules, to create an executable algorithm.

The naïve rules to describe the airflow were derived using a large data set of airflow patterns within the environment. These patterns were derived by running a range of simulations (using fluid dynamics software called Flo++), and key 'real world' experiments to: (a) verify simulation accuracy, and (b) ascertain the limitations of the simulations. The most important feature of the naïve rules is that they must be capable of modeling the airflow sectors.

The naïve rules are encapsulated in an algorithm implemented with C++. It simulates actual flows, governed by the naïve rules. A flow is represented by a list of flow points, called FD (Flow Data) points. As the flow continues in a straight line, the position of the final FD point is advanced. If the flow is deflected (undergoes a change in direction), then a new flow point is added to the list. Each FD point has a position, direction and momentum (represented by a vector). The direction describes the angle of the preceding segment. The momentum is used to calculate the position of the following segment.

Finally, the airflow modeling using naïve physics have a series of procedure. First, observe physical process; second, derive naïve rules to describe physical process; thirdly, create algorithm that encapsulates the rules; lastly, implement the algorithm.

In addition, in 2009, the author did some improvement in the papers Improving the robustness of naive physics airflow mapping, using Bayesian reasoning on a multiple hypothesis tree and Bi-modal search using complementary sensing (olfaction/vision) for odor source localization.

B. Reasoning system for source prediction.

The Reasoning System is a ‘Sense-Map-Plan-Act’ control scheme that emulates common sense style behavior. It is comprised of four phases.

The author used a robot to conduct experiments. The robot used for locating the source of an odor and investigating the airflow and odor conditions is named Roma. Roma is equipped with several sensors: a wind vane chemical sensor and bump sensor. The wind vane measures the airflow direction. The chemical sensor has a conductivity that is dependent on the concentration of reducing gasses

The algorithm creates a list of candidates (potential odor sources), and a list of target positions. First, the candidates are identified: they include positions on objects, or the inlet. Legitimate flows and sectors that would contain odor from each candidate are identified as targets, and associated with that candidate. To be legitimate, a sector must be above a minimum size (area), and a flow must be above a minimum length. Using the list of candidates, a list of targets is compiled, where each target is associated with a unique set of candidates, so that there is at least one target for each candidate, and no redundant targets.

The robot then moves to each of these target positions, governed by the path planning strategy (expanded below). The algorithm attempts to minimize the total distance moved, which is a form of the traveling salesman problem. The simple heuristic ‘move to the next closest target’ is applied. With this strategy, the distance usually appears to be close to minimum; and otherwise, the penalty is not significant. At each position, the chemical concentration is sensed and recorded, the next target position is found, and the process repeated.

After the robot has visited all relevant targets, the target list is sorted by descending chemical concentration. The list is segmented into groups of similar concentration. The odor source is predicted to be one of the candidates associated with the first group. The confidence of the prediction is determined by the position of the greatest disparity between segments.

Path planning is divided into two levels of control: reactive bump response and path modification. Given the robot’s current position and target position, the path is determined by calculating two paths parallel to and on either side of, the direct path. If

there is a predicted collision, then an intermediate position is proposed to avoid the collision. The process is repeated recursively for the new path to a depth of four.

C. Experimental analysis

The Australia group did some experiment to demonstrate their papers. The following experiments were conducted: The airflow and odor concentration conditions were investigated, a conventional reactive odor localizing robot was tested, and the odor localization system reported in this paper was tested. The environment and experiments are explained in this section.

The experiments were conducted with a small mobile robot named Roma. The experiments were conducted in a pseudo-2D mini 'world'. Roma was placed in the 'mini' world with a number of objects, and then commanded to locate the odor source, which was placed at the inlet or within one of the objects. The results demonstrate that this method is relatively robust and general.

The group's main member is Hiroshi Ishida, whose research interests are in biomimetic electronics with emphasis on chemical sensors and their applications in robotics. They have proposed using autonomous mobile sensing system to locate the odor source in a clean room.

The probe with gas and anemometric sensors was used to determine an odor-source direction. Four thermistor anemometric sensors located around a pillar were used to determine the wind direction. The eight normalized patterns measured at every 450 of the wind direction and the one with equal sensor outputs were stored in the computer. The wind direction is determined by selecting the pattern with the minimum Euclidean distance to the measured one.

In addition to the anemometric sensors, four semiconductor gas sensors were placed at the vertices of a square. The output of a gas sensor, S, is defined as the ratio of the sensor resistance in gas, Rgas, to that in air, Rair. The relationship between S and vapor concentration C is almost linear while the concentration is low. The sensor responses were calibrated by determining the positive constant a in the following equation $S = 1 – aC$.

This linearity leads to a simple calculation of the concentration gradient vector G from the output differences of two pairs of the gas sensors located in diagonal vertices.

The most effective algorithm to localize an odor source in the wind tunnel was the combination of upwind searching and moving across the wind according to the gas-concentration gradient. However, this exploratory algorithm was not always effective in the clean room. The author gave a new algorithm with two strategies for the inside and for the outside of the plume. The algorithm is separated into five phases.

Phase 1: Waiting for gas detection

Odor-source localization is made to start when the presence of the target gas is detected. Setting a threshold eliminates the baseline drift of the gas sensors and used for determination of the presence of gas.

Phase 2: Searching for plume along concentration gradient

The stage is made to move according to the gas-concentration gradient G when the gas concentration is low. As the gradient across the wind is relatively steep, the stage approaches the central axis of the plume.

Phase 3: Retreat

When the target gas disappears, the stage is made to backtrack. The phase is changed back to Phase 2 when the target gas is detected again. If the stage reaches the point where the average of the four gas sensor outputs was minimized in the previous track, the stage is made to stop and the phase is changed to Phase 1.

Phase 4: Tracking plume

If the gas concentration is high enough, a gas source is expected to be situated in the upwind direction and, thus, upwind search is effective in the plume. The concentration gradient across the wind direction is also used to keep the stage heading toward the center of the plume.

Phase 5: Searching for plume across wind

When the plume is lost, the stage is made to move to and fro across the wind. The map described in Phase 3 is used to gradually expand the scanning width. If the gas sensor response becomes below the threshold, the phase is changed to Phase 4 since the stage is thought to reenter the plume. When the plume is not found after searching at right and left hand twice respectively, the plume is regarded as completely lost and the phase is changed to Phase 2.

The experiments were conducted in a clean room. The author has done three experiments, which are track of stage moving to upwind direction, track of stage moving along gas-concentration gradient and track of stage by full set of new algorithm.

There are many useful and humanitarian applications for robots that can locate the source of a chemical plume in an open area free of obstacles. The Australia group used naïve physics for odor location in a cluttered indoor environment, while the Japanese group has used a 'combination of plume acquisition and following'. They use absolute readings from wind direction and chemical sensors.

Australia group used naïve physics to model the airflow in the robot's environment. Then a reasoning system uses the airflow model to propose the path of an odor plume and predict the most probable locations of the odor source. This approach has been shown to be effective for odor localization in a known indoor environment, without the need for the robot to travel to the source.

Japanese group modeled the plume by fitting a chemical distribution model to the sensor dynamics (Ishida et al., 1997). This is reliable for source detection. However, the robot must continually re-calculate and alter its heading, making it partially reactive, and the robot must travel to within 20 cm of the source. The methods largely focus on basing robot sensing and algorithms on the odor localizing behavior, aiming for simplicity using reactive control schemes and local sensing. The main limitations are that robots must follow the plume along its entire length, which is time consuming and may not be possible. In addition, the effects of obstructing objects and walls on robot mobility are often neglected.

Comparing the two group, the Australia group using the qualitative modeling, and the Japanese group using the quantitative approach. The Australia group did the experiment in an indoor environment, while the Japanese group in a clean room where there are always slight wind produced by air conditioner.

1.3. THE STRUCTURE OF THE DISSERTATION

In this dissertation, there are seven sections. The first section is literature review that talks the recent research. The second section is about particle paths mapping. Using interpolation and extrapolation methods, chemical particle paths mapping demonstrates the information of airflow and chemical particle at every position in the area of interest.

The third section is compare and validate about interpolation and extrapolation approach. The fourth section shows the results of simulations and real world experiments. In the fifth section, the algorithm of multi-sensor fusion is proposed. It can smooth the chemical particle paths by computing more sensor data. The updated mapping has the better precision than the old one. In the sixth section, reasoning system and control scheme are implemented, aiming to detect and localize the maximum likely-hood of chemical source. At the last section, it’s the conclusion and the future work.

2. USING INTERPOLATION AND EXTRAPOLATION METHOD TO GET ODOR PROPAGATION FOR AIRBORNE CHEMICAL SOURCE DETECTION

This section addresses the problem of mapping likely particle path derived from a chemical source using interpolation and extrapolation method. Order localization is the problem of finding the source of an odor or other volatile chemical. Most localization method require the robot to follow the odor plume along its entire length, which is time consuming and may be especially difficult in a cluttered environment. In this section, a map of sensors' environment was used, together with the path line of airflow, to predict the pattern of air movement. The robot then used the airflow pattern to reason about the probable location of the odor source. This demonstrates that interpolation and extrapolation method can be used to assist odor localization search and indicates that similar techniques have great operating in an unstructured environment to reason about its surroundings. The author will present details of getting the model of particle path using interpolation and extrapolation method, model of particle path surrounding the obstacles and openings, result of practical odor source location simulation.

2.1. INTRODUCTION

The detection of the airborne chemicals presents a different type of challenge than the more traditional detection efforts, such as visual-based detection or propagating signal detection. The chemicals that are airborne tend to drift in various directions due to wind, up-draft, and obstacles. As a result, isolation of the source of such particles becomes considerable difficult and dependent on topography and environment.

There has been some previous research on the detection and modeling of airborne particles, plume location and tracking. However, most of such research is based on sensor information on moving robots that are guided by the detectors. These types of sensing robots are assumed to move about freely following the trail of a chemical signature, while they're continuously sensing for the particles. Both of these assumptions are not valid in inaccessible and hostile environments with sensors that can either function once or need along rejuvenation time cycles.

In our approach to the problem of chemical particle detection and source location, the author use a small number of chemical sensors that are sparsely scattered around an

area only known by a two-dimensional map. In real-world problems, the author anticipate that a small unmanned aircraft would drop some of these sensors on the area of interest while taking some aerial pictures. the author assume that the sensor data along with the map are transmitted to a nearby location perhaps to a vehicle that will be traveling through the area of interest. The author would like to use the maximum available information content to generate first a model of the chemical particle distribution, and then locate the source of the particles based on the model.

2.2. PROPERTIES OF CUBIC SPLINE INTERPOLATION

In the computer science subfields of computer-aided design and computer graphics, the term spline more frequently refers to a piecewise polynomial (parametric) curve. Splines are popular curves in these subfields because of the simplicity of their construction, their ease and accuracy of evaluation, and their capacity to approximate complex shapes through curve fitting and interactive curve design.

The spline curve was constructed by using a different cubic polynomial curve between each two data points. In other words, it is a piecewise cubic curve, made of pieces of different cubic curves glued together. The pieces are so well matched where they are glued that the gluing is not obvious. Cubic spline interpolation is a special case for Spline interpolation that is used very often to avoid the problem of Runge's phenomenon. This method gives an interpolating polynomial that is smoother and has smaller error than some other interpolating polynomials such as Lagrange polynomial and Newton polynomial.

Suppose that $\{(\mathrm{x}_i, y_i)\}_{i=0}^{N}$ are N+1 points, where $a = x_0, x_1, ..., x_n = b$ .The function S(t) is a real piece-wise function on the interval [a, b] composed of N subintervals $[x_{i-1}, x_i]$. The restriction of S(t) to an interval I is a polynomial $P_i : [\mathrm{x}_{i-1}, x_i] \to \mathbb{R}$. So that,

$$
\begin{aligned}
&S(\mathrm{x}) = \mathrm{P}_1(\mathrm{x}), \quad \mathrm{x}_o \le x \le x_1, \\
&S(\mathrm{x}) = \mathrm{P}_2(\mathrm{x}), \quad \mathrm{x}_1 \le x \le x_2, \\
&\vdots \\
&S(\mathrm{x}) = \mathrm{P}_n(\mathrm{x}), \quad \mathrm{x}_{n-1} \le x \le x_n.
\end{aligned}
$$

Theorem 1: Giving any two sensors' coordinate location (x_{i-1}, y_{i-1}) and (x_i, y_i), the particle path can be described by cubic hermit spline function $P_i : [x_{i-1}, x_i] \to \mathbb{R}$. And the parameters of the function are unique.

Proof:

Each segment of the cubic polynomial function is a cubic polynomial:

$$p(t) = at^3 + bt^2 + ct + d = [t^3\ t^2\ t\ 1][a\ b\ c\ d]^T$$

Two dimensional polynomials, one for each coordinate:

$$x(t) = a_x t^3 + b_x t^2 + c_x t + d_x$$
$$y(t) = a_y t^3 + b_y t^2 + c_y t + d_y$$

The functions should satisfy endpoints and tangents constraints

$$p(0) = p_1 = (x_1, y_1) \quad p(1) = p_2 = (x_2, y_2)$$
$$p'(0) = \bar{p}_1 = (\bar{x}_1, \bar{y}_1) \quad p'(1) = \bar{p}_2 = (\bar{x}_2, \bar{y}_2)$$

The four constraints can solve the four unknowns. In the matrix formulation,

$$[x\ y] = [t^3\ t^2\ t\ 1] \begin{pmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \\ \bar{x}_1 & \bar{y}_1 \\ \bar{x}_2 & \bar{y}_2 \end{bmatrix}$$

Theorem 2: Let the author assume f be the actual particle path defined on $[a,b]$, with $a = x_0, x_1, ..., x_n = b$, and let S be the cubic spline interpolant of f. The cubic spline has minimum curvature property $\int_a^b [s''(x)]^2 dx \le \int_a^b [f''(x)]^2 dx$.

Proof:

The curvature of f is $k(x) = \dfrac{|f''(x)|}{(1+|f'(x)|^2)^{3/2}} \approx |f''(x)|$

Hence, $\int_a^b [f''(x)]^2 dx$ is a crude measure of the total curvature over an interval.

Algebra identity: $F^2 - S^2 = (F-S)^2 - 2S(S-F)$.

Let $F = f''(x)$ and $S = s''(x)$

$$\int_a^b [f''(x)]^2 dx - \int_a^b [s''(x)]^2 dx = \int_a^b [f''(x) - s''(x)]^2 dx - 2\int_a^b s''(x)(s''(x) - f''(x)) dx$$

The first term is ≥ 0, the author will show that the second term is 0.

$$\int_a^b \mathrm{s}''(x)(\mathrm{s}''(x)-\mathrm{f}''(x))dx = \sum_{i=0}^{n-1}\int_{x_i}^{x_{i+1}} \mathrm{s}''(x)(\mathrm{s}''(x)-\mathrm{f}''(x))dx$$

$$= \sum_{i=0}^{n-1}(\mathrm{s}''(x)(\mathrm{s}''(x)-\mathrm{f}''(x))\Big|_{x_i}^{x_{i+1}} - \int_{x_i}^{x_{i+1}} \mathrm{s}'''(x)(\mathrm{s}''(x)-\mathrm{f}''(x))dx)$$

$$= \sum_{i=0}^{n-1}(-\int_{x_i}^{x_{i+1}} \mathrm{s}'''(x)(\mathrm{s}''(x)-\mathrm{f}''(x))dx)$$

Telescoping sum and $\mathrm{s}'''(x)$ is constant, and using $s(x_i) = f(x_i), i = 0,...,\mathrm{n}.$ the second term is equal to 0.

Theorem 3: let us assume the author use a parametric representation of a cubic spline curve to represent the particle path, $s(t) = (x(t), y(t)),$ then, the shortest cubic spline interpolation between two sensor points (x_{i-1}, y_{i-1}) and (x_i, y_i), is

$$\min L = \sum_{i=0}^{n-1}\int_{t_i}^{t_{i+1}} \sqrt{(x'(t))^2 + (y'(t))^2}\,dt$$

.

Proof:

The arc length of each spline segment on the curve is

$$l_i = \int_{t_i}^{t_{i+1}} \sqrt{(x'(t))^2 + (y'(t))^2}\,dt \quad (i = 1,2,...,n)$$

Thus, the arc length of the whole curve is $L = \sum_{i=0}^{n-1} l_i$.

The arc length of the curve $s(t)$ can be approximation by the composite Simpson rule. The function is about $s(t_0) = (\mathrm{x}(t_0), \mathrm{y}(t_0))$ and $s(t_{n-1}) = (\mathrm{x}(t_{n-1}), \mathrm{y}(t_{n-1}))$ .the problem of the shortest cubic spline interpolation is to find suitable $s(t_0)$ and $s(t_{n-1})$.

Error estimation. Now the author assume that the interpolation node (x_n, y_n), are points from a smooth plane curve $\bar{C} = \{(u(s), v(s) : 0 \leq s \leq S; \mathrm{u}, \mathrm{v} \in C^2[0, S]\}$, where s denotes a certain parameter, with the partition $\Pi_S : 0 = s_0 < s_1 < ... < s_n$.

The interpolation nodes then read as $u(s_n) = x_n, v(s_n) = y_n$.

By $C = \{(x(t), y(t)) : 0 \leq t \leq T\}$, the author denote the interpolating curve, where x, y are polynomial splines. Each point if the curve $\bar{C}$ may be written in the form,

$$(u(s_{n-1} + \sigma h_n), v(s_{n-1} + \sigma h_n),\ 0 \leq \sigma \leq 1,\ h_n = s_n - s_{n-1.}$$

In the same manner the author write the points of C in the form

$(x(t_{n-1}+\sigma k_n), y(t_{n-1}+\sigma k_n),\ 0\le\sigma\le 1,\ k_n = t_n - t_{n-1.}$.

Since C interpolates $\bar{C}$, the points $(x(t_n), y(t_n))$ and $(u(s_n), v(s_n))$ coincide.

In each subinterval the author associate the point $(u(s_{n-1}+\sigma h_n), v(s_{n-1}+\sigma h_n)$ of the curve $\bar{C}$ with the point $(x(t_{n-1}+\sigma k_n), y(t_{n-1}+\sigma k_n)$ of curve C. So the author get a mapping between the parameters S and t, $(t-t_{n-1})/k_n = (s-s_{n-1})/h_n$, $(t\in[t_{n-1},t_n], s\in[s_{n-1},s_n])$, and the functions u and v become functions of the parameter t:

$$u(s) = u(s_{n-1}+(t-t_{n-1})h_n/k_n = \tilde{u}(t),$$
$$v(s) = v(s_{n-1}+(t-t_{n-1})h_n/k_n = \tilde{v}(t).$$

Using the error estimation separately for $x(t)-\tilde{u}(t)$ and $y(t)-\tilde{v}(t)$ we get,

$|x(t)-\tilde{u}(t)|\le C^x k^2$, $|y(t)-\tilde{v}(t)|\le C^y k^2$,

with $C^x = \frac{3}{8}M_2^x + TM_3^x$, $C^y = \frac{3}{8}M_2^y + TM_3^y$, $k = \max(t_n - t_{n-1})$,

where M_i^x, M_i^y are bounds for the i-th derivatives of $\tilde{u}(t)$ and $\tilde{v}(t)$.

Note that even in the case $C=\bar{C}$ we may not expect generally the equalities $|x(t)-\tilde{u}(t)|=0$ and $|y(t)-\tilde{v}(t)|$=0, however, the errors are bounded.

2.3. DESCRIPTION OF THE METHODOLOGY

As a test case, the author consider a three sensor configuration system as in Figure 2.1. In the figure, the thick black lines are the boundaries of the room, the red dots are the sensor locations, and the red dotted lines designate the border of the boundary zone.

Some chemical sensors are designed to detect simply the existence of a chemical particle and trigger a positive result when the concentration amounts are above a preset threshold level. In our design, instead of the threshold, the author make use of the actual concentration levels that are detected. This approach along with some other data enables us to model the flow of the particles and the location of the source. Each sensor provides the co-located sensory information of the wind velocity, the concentration of the particles, and the concentration differential preferably perpendicular to the wind direction. The concentration differential information is obtained not by an additional

sensory device but by an off-centered multi-orifice detection hardware configuration. In our derivations, the author assume that the differential information is perpendicular to the wind direction, but the author can accommodate any non-zero known angular orientation simply by a coordinate transformation. Designating the location of the sensors by (x, y), the author represent the flow of air by $\delta x, \delta y$. Similarly, the author represent the sensed particle concentration by s and the concentration gradient by δs.

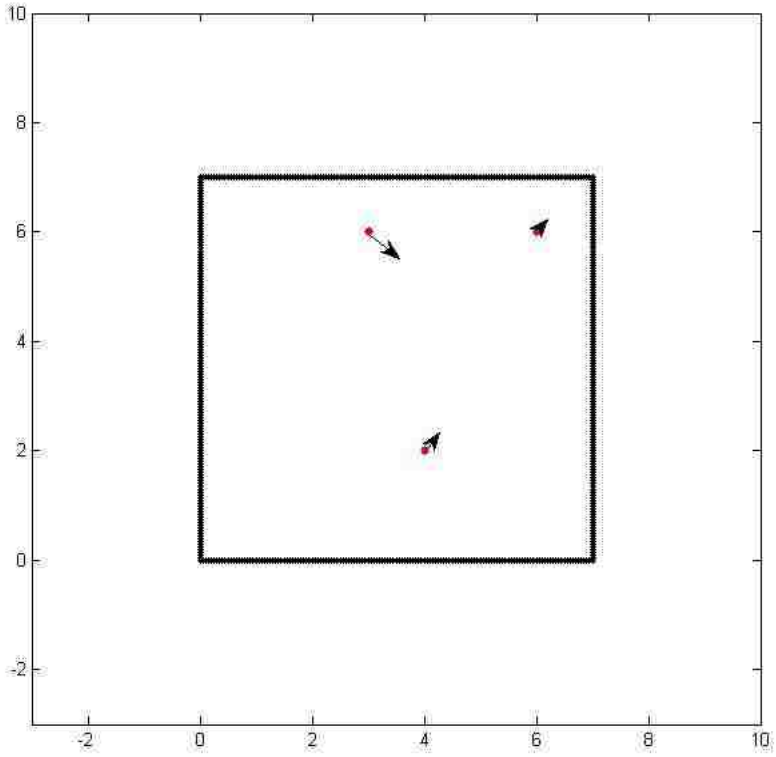


Figure 2.1. The location of three sensors in a square enclosure.

Once the author obtain the sensory information, the author start with an approximation of the particle path. In order to avoid multiple solutions, the author make a number of assumptions.

Assumption 1: The air-borne particles travel the most direct route.

Based on Assumption 1, the author configure paths that go through the sensor locations, such that the paths satisfy the locations as well as the differentials. This approach leads to a parametric cubic-polynomial representation of the path in terms of a variable t. the author use the cubic Hermite splines with the end point differentials weighted three times, such that

$$\begin{aligned} x(t) &= (2(x(0)-x(1))+3(\delta x(0)+\delta x(1)))t^3+3(x(1)-x(0)) \\ &\quad -3(\delta x(1)+2\delta x(0)))t^2+3\delta x(0)t+x(0), \\ y(t) &= (2(y(0)-y(1))+3(\delta y(0)+\delta y(1)))t^3+3(y(1)-y(0)) \\ &\quad -3(\delta y(1)+2\delta y(0)))t^2+3\delta y(0)t+y(0). \end{aligned}$$

where the parametric curve starts at one sensor location at x(0), y(0) and ends at the other sensor location at x(1), y(1) as t goes from 0 to 1. Figure 2.2 shows the spline approximation of a particle path from one sensor to another with matching initial and final velocities.

If the author strictly apply this interpolation method, the author end up two possible choices, one path going from Sensor 1 to Sensor 2 and another path going from Sensor 2 to Sensor 1. The author could explore both possibilities or have a decision making process based on other factors to eliminate one of the choices. Here, as a first step, the author choose the shortest path option.

Assumption 2: The path of the air-borne particles has minimal length.

Even though, the author now have a path from one sensor to another, there are still couple of issues to be resolved. The first issue is related to the underlying presumption that a particle somehow would travel from one sensor to the other even though the sensors are at arbitrary locations. To correct this problem, the author rely on the dissipation property of the particles. The author compute the expected concentration value along the computed path and compare it with the actual sensed concentration value. Based on the error and the measured gradient concentration, the author determine a new location perpendicular to the initial path where the expected and sensed concentration

values match. The author then compute the corrected path going through one of the sensors and the new location. When the author repeat the process forwards from one sensor and backwards from another one, the author end up getting two consistent paths with correct concentration values.

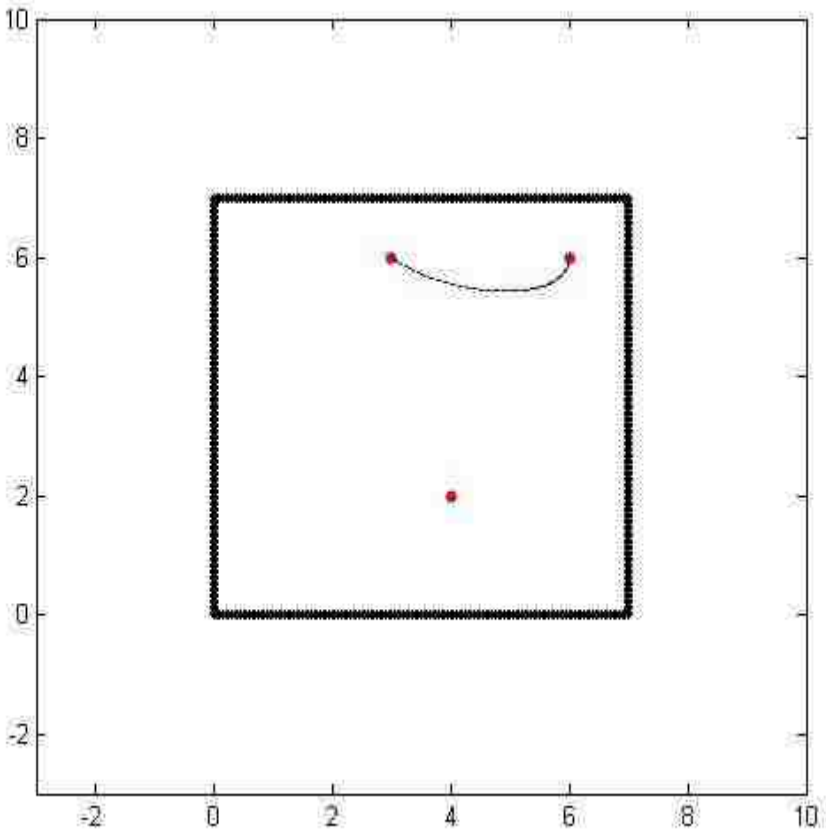


Figure 2.2. An air-borne particle path with matching terminal velocities.

The second issue is related to the choice of the parameter t. In our parametrization, the author chose t to start at 0 at one of the sensors and end at 0 initially

at another sensor at 1. The author would like to have the parameter be a good representative of actual travel time, since the author also would like to obtain connected paths. Figure 2.3 shows the two paths generated by matching the expected and sensed concentration values.

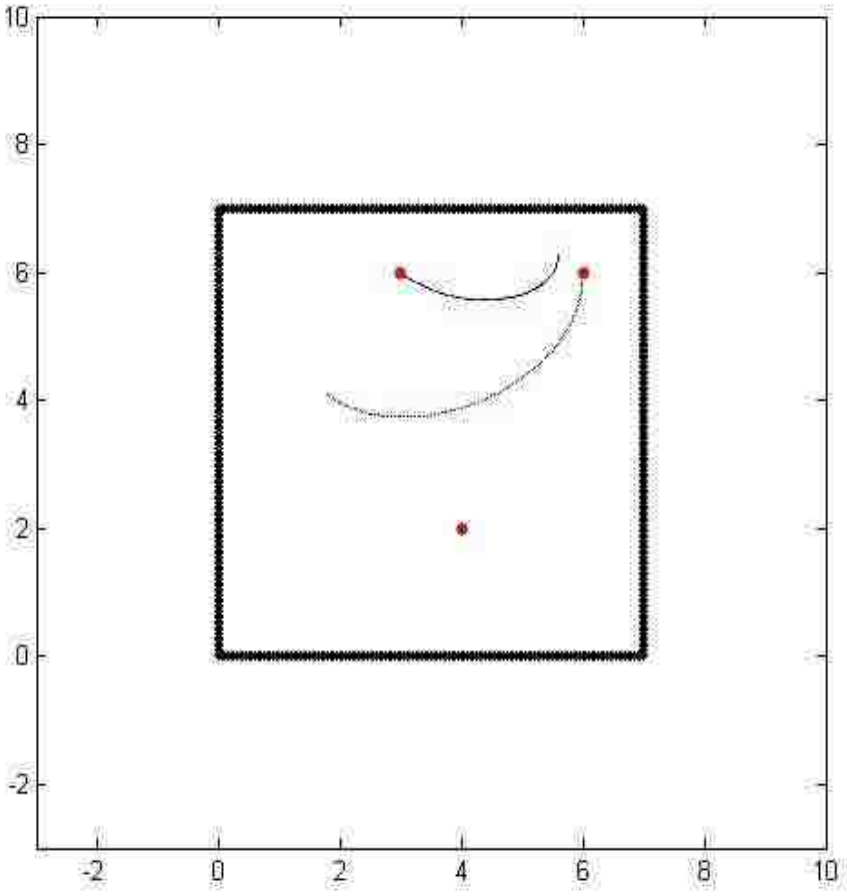


Figure 2.3. Consistent air-borne particle paths between two sensors.

To correct this problem, the author compute the speed at every point along the path as a linear function of the distance from one sensor to the other one while matching the sensed speed values at the two end points. In the existence of multiple sensors, the author need to make sure that the paths continue smoothly passing through the neighborhood of sensors. Figure 2.4 shows the two paths with equally-timed distances.

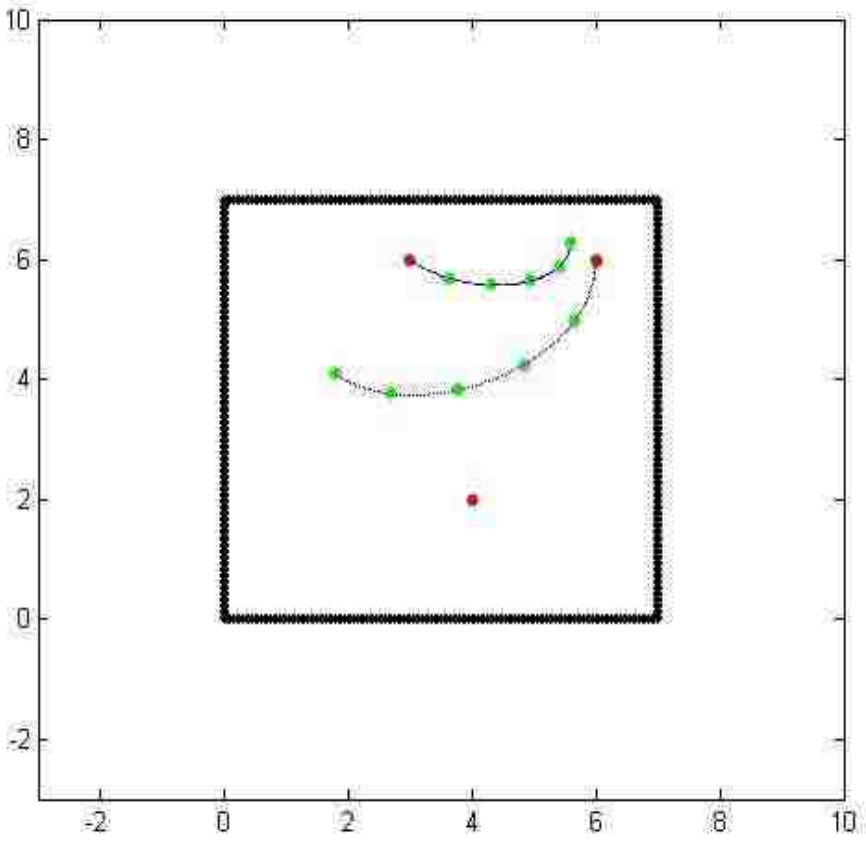


Figure 2.4. Paths with equally-timed distances among two sensors.

In this example, this extended path goes through Sensor 1 and the neighborhoods of Sensors 2 and 3 due to path updates for the consistency of the particle propagation. To obtain a general development of all possible paths passing through the neighborhoods of sensors, the author introduced a couple of more assumptions.

Assumption 3: The individual segments of the paths that span multiple sensor regions have to have the same flow directions.

Assumption 4: The paths that span multiple sensor regions always go through at least one sensor.

Based on above assumptions, Figure 2.5 shows the curves that go through the three sensor regions, and Figure 2.6 shows the same paths with equally-timed distances.

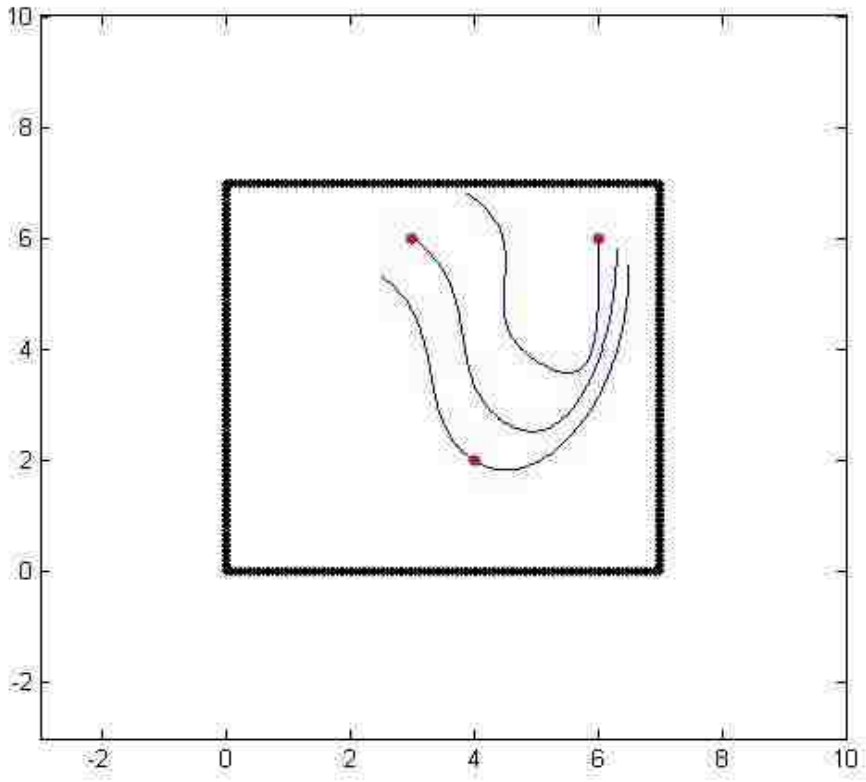


Figure 2.5. Consistent air-borne particle paths among three sensors.

As the author observe from the Figures 2.4 and 2.6, there're are two set of possible particle flow directions. At this point, the author pursue both possibilities and make a decision on the actual flow path later by a conflict resolution layer.

As the author described above, in the process of determining the particle paths that go through the sensors, the author primarily rely on the sensory values, the particle dissipation properties, and interpolation. However, in order to generate a full coverage of paths for the whole room, the author need to extrapolate as well.

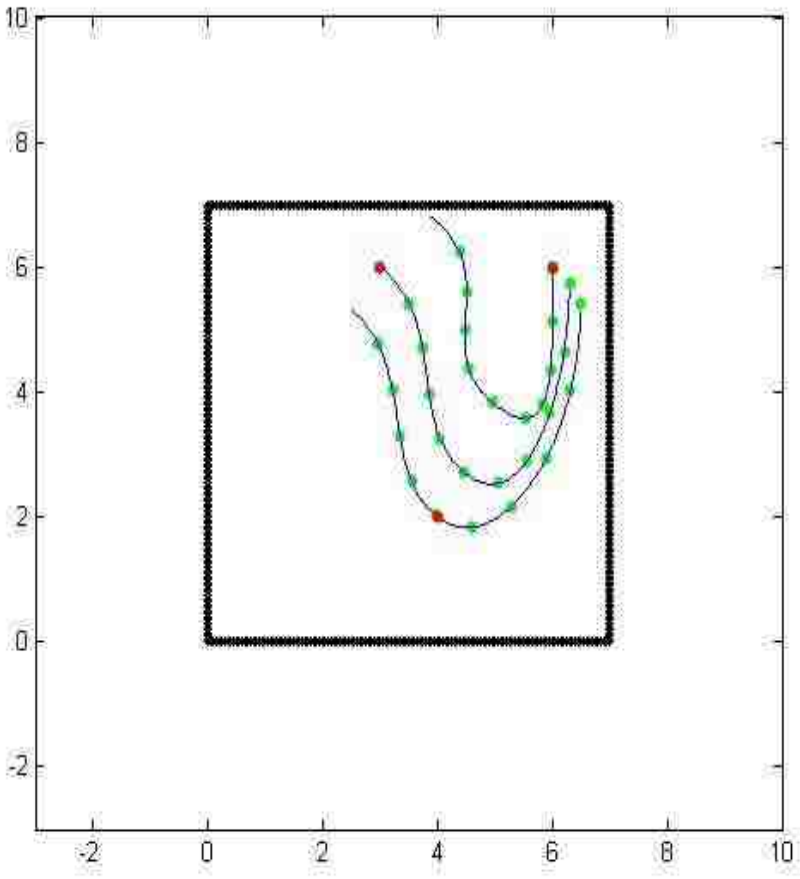


Figure 2.6. Paths with equally-timed distances among three sensors.

In the first step of the interpolation, the author extend the primary paths that go through the sensors with linear approximations of the air flow parameters, and the particle dissipation properties of the particle concentrations. Figures 2.7 and 2.8 show the path extensions, where the author purposely extended the paths beyond the room boundaries. The author will take care of the portions that go beyond the boundaries especially after the whole room coverage below.

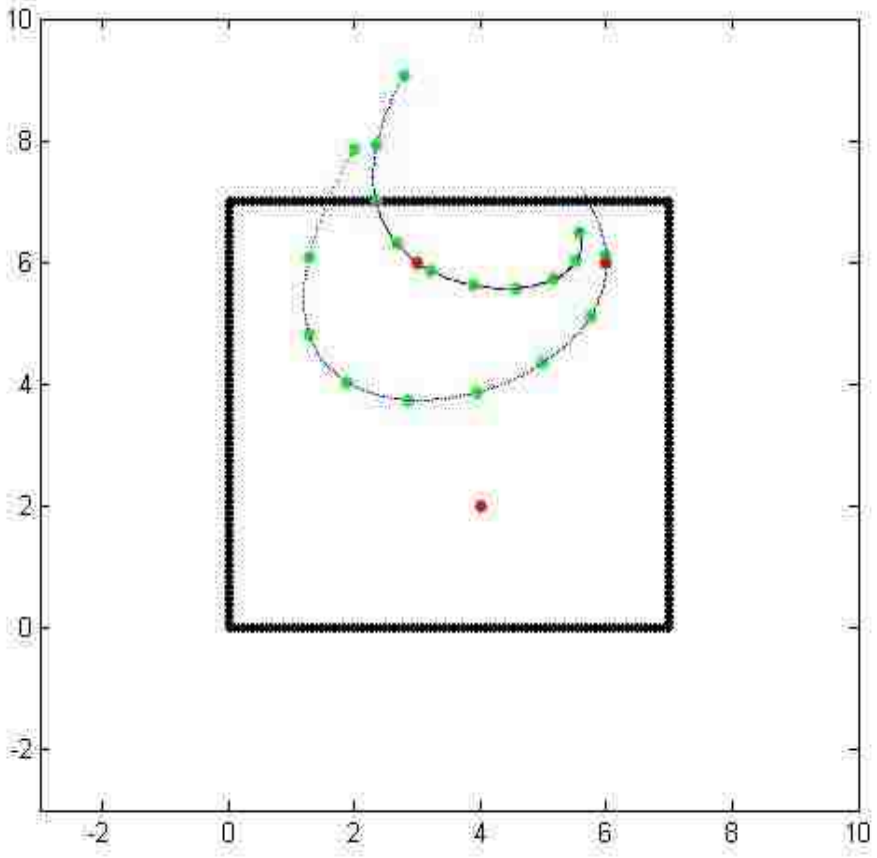


Figure 2.7. Primary air-borne particle path extensions going through two sensors.

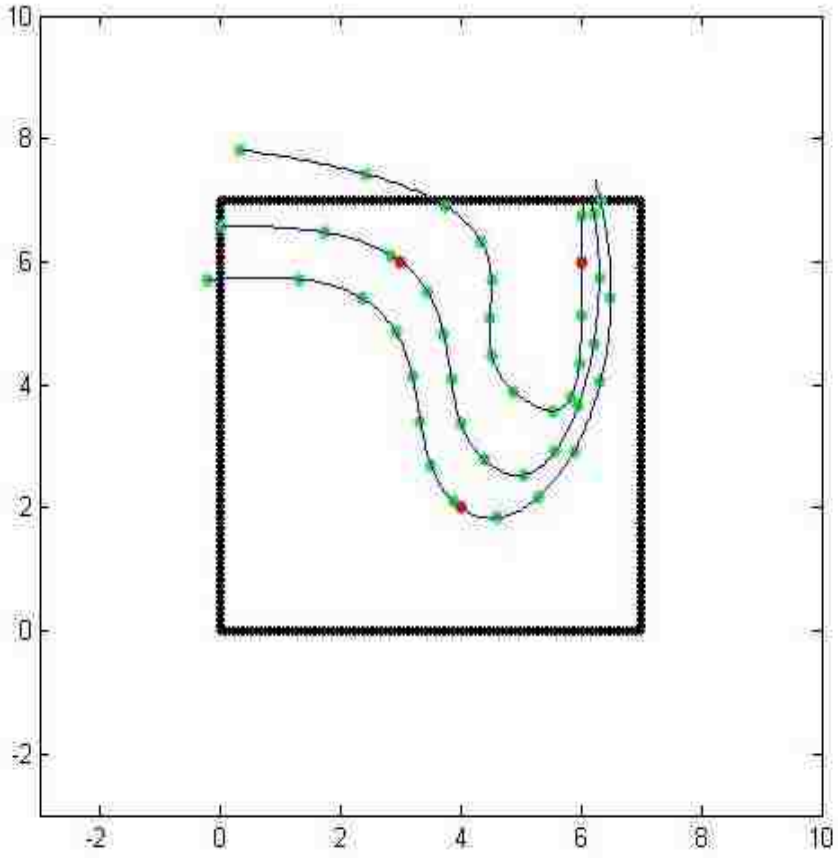


Figure 2.8. Primary air-borne particle path extensions going through three sensors.

In the next step of the extrapolation, the author fill the whole room with secondary paths.

For the secondary paths that are between two adjacent primary paths, the author determine the normals (perpendicular lines to the tangents of the paths), and use the intersection points of the normals to generate a secondary path. The author assign the average values of the particle concentrations and the concentration gradients on these paths.

For the secondary paths that are on the outside regions of the primary paths, the author use similar normal extensions, but the author extrapolate the particle

concentrations and the concentration gradients. Figures 2.9 and 2.10 shows the path extensions as well as the whole room coverage with primary and the secondary paths.

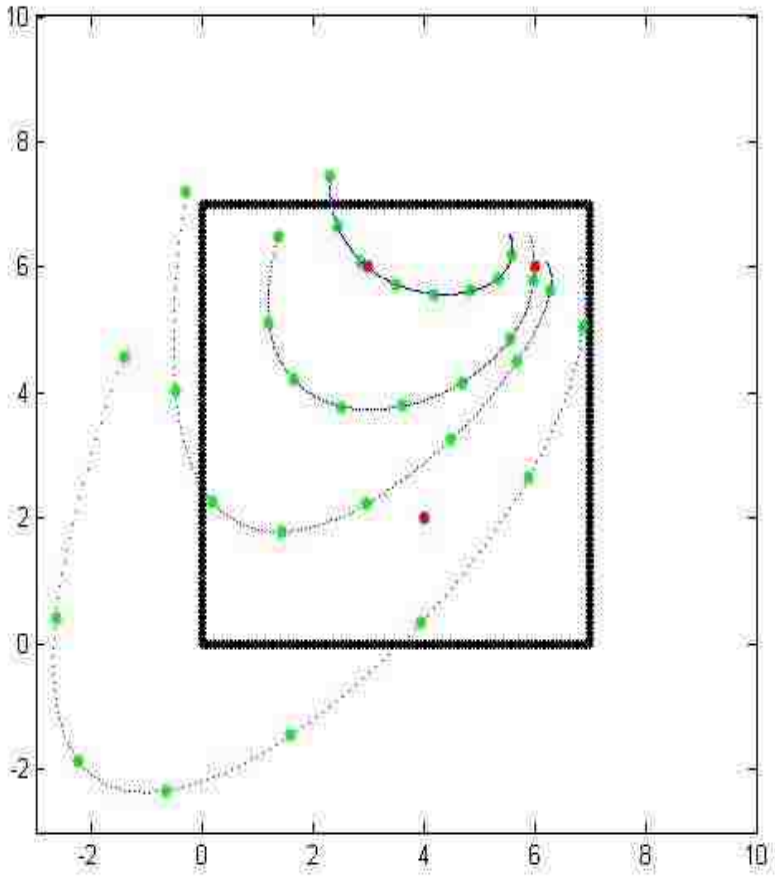


Figure 2.9. Primary and secondary air-borne particle paths going through two sensors.

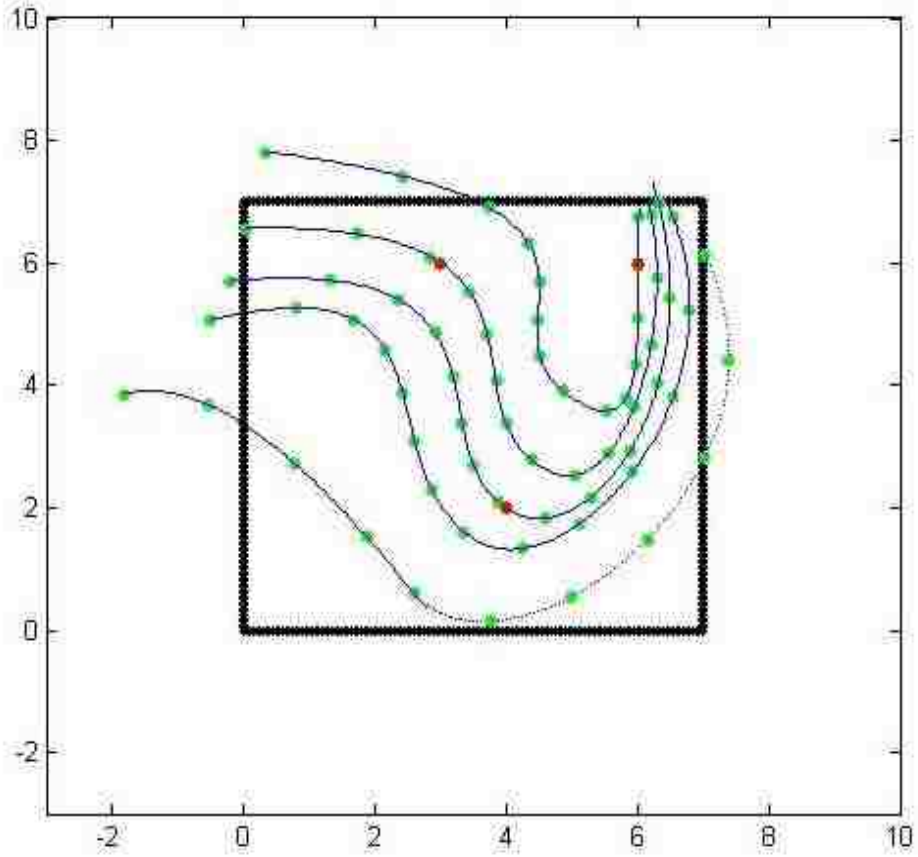


Figure 2.10. Primary and secondary air-borne particle paths going through three sensors.

The extended primary and the secondary paths depend on the best data fit based on the sensory data. In the generation of these paths, the author excluded the effects of the room boundaries. Obviously, if the author truncate the paths at the boundary, we'll generate unnatural particle behavior, where particle would originate on one side of the boundary and disappear on the other side. To model the particle path along these types of boundary structures, the author need to incorporate realistic flows of air in the neighborhood of the boundaries.

If the author denote the perpendicular distance of a point on one of the paths by d, then the author compute the new distance $\hat{d}$ as $\hat{d} = w\tanh(d/w)$, where w is the width of the buffer zone. This approach maps the paths at infinity onto the room boundary and enables all of the portions outside the room boundary to be in the buffer zone. Figures 2.11 and 2.12 show the complete set of paths confined within the room bound

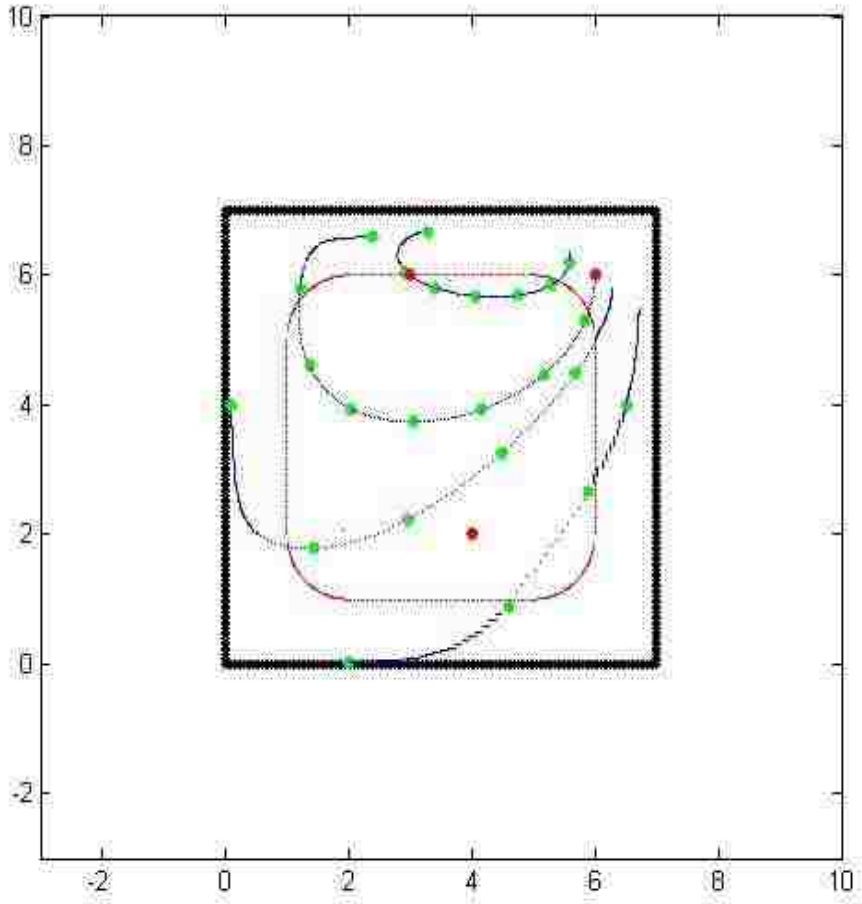


Figure 2.11. Boundary confined air-borne particle paths going through two sensors.

Our approach to handle the boundaries is based on a convex buffer zone in the neighborhood of the room boundary, where the air-borne particle flow is distorted in the direction of the boundary. In this zone, the author map the portions of the paths, which are in the buffer zone or outside the room boundary, in a manner that preserves the continuity of the location and the derivative of a particle entering this zone.

Assumption 5: The paths that go across the boundary regions always go close to the boundary infinitely.

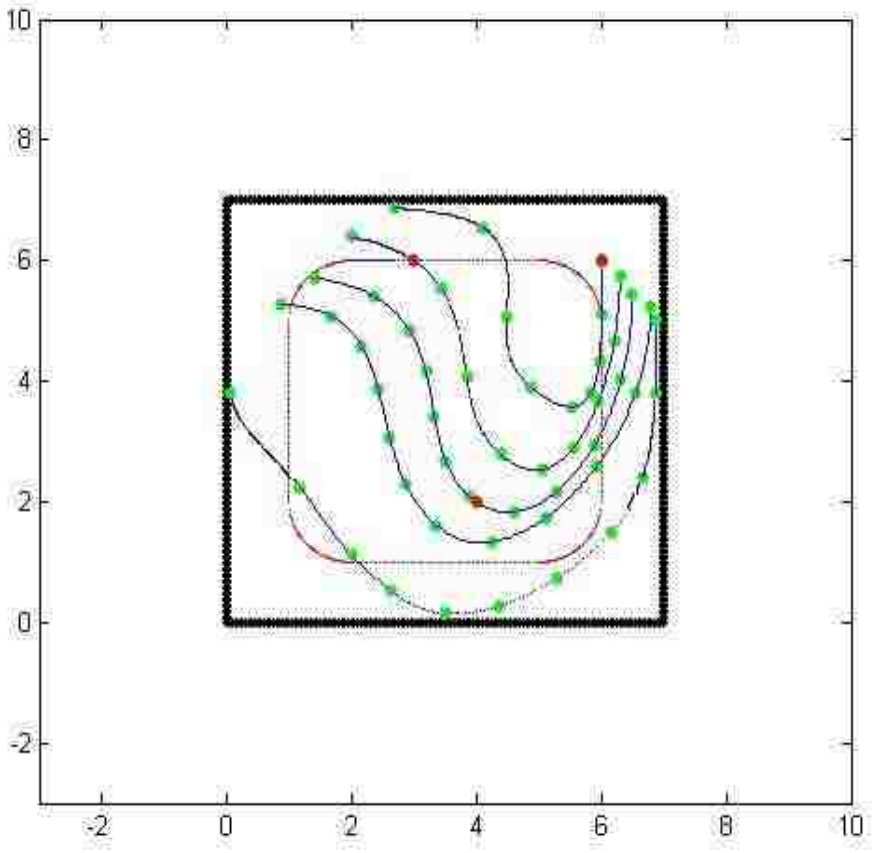


Figure 2.12. Boundary confined air-borne particle paths going through three sensors.

To further validate our approach, the author applied the same principles on a slightly larger room in Figure 2.13 and with twice the air speed in Figure 2.14. As the author can observe from these figures, our approach provides satisfactory and consistent air-borne particle flow paths based on few sensory data.

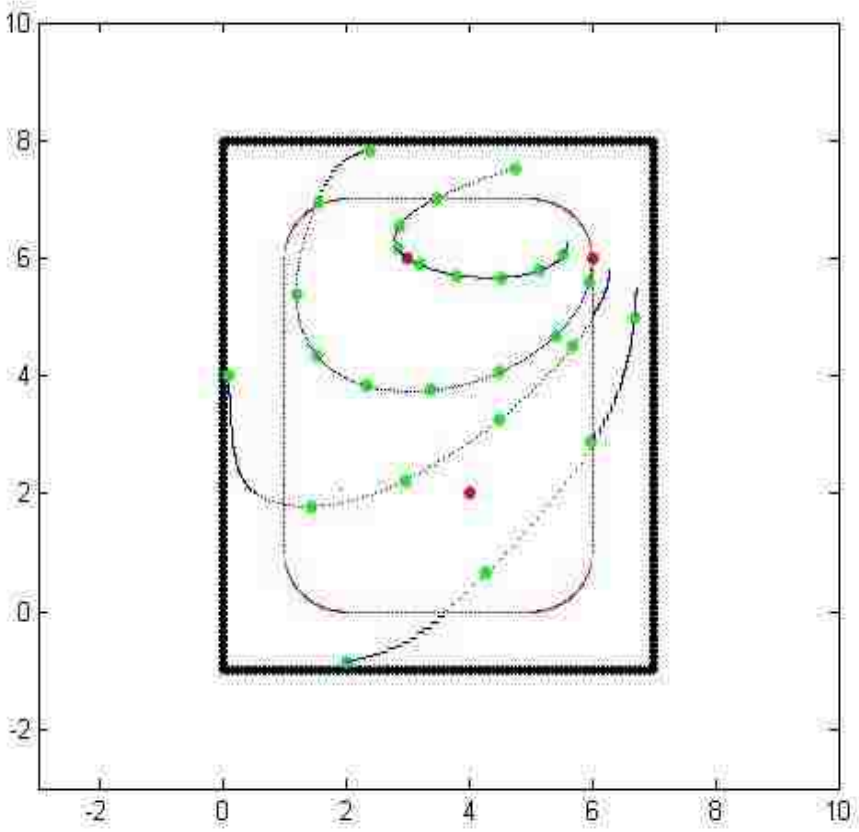


Figure 2.13. Boundary confined air-borne particle paths going through two sensors in a rectangular room.

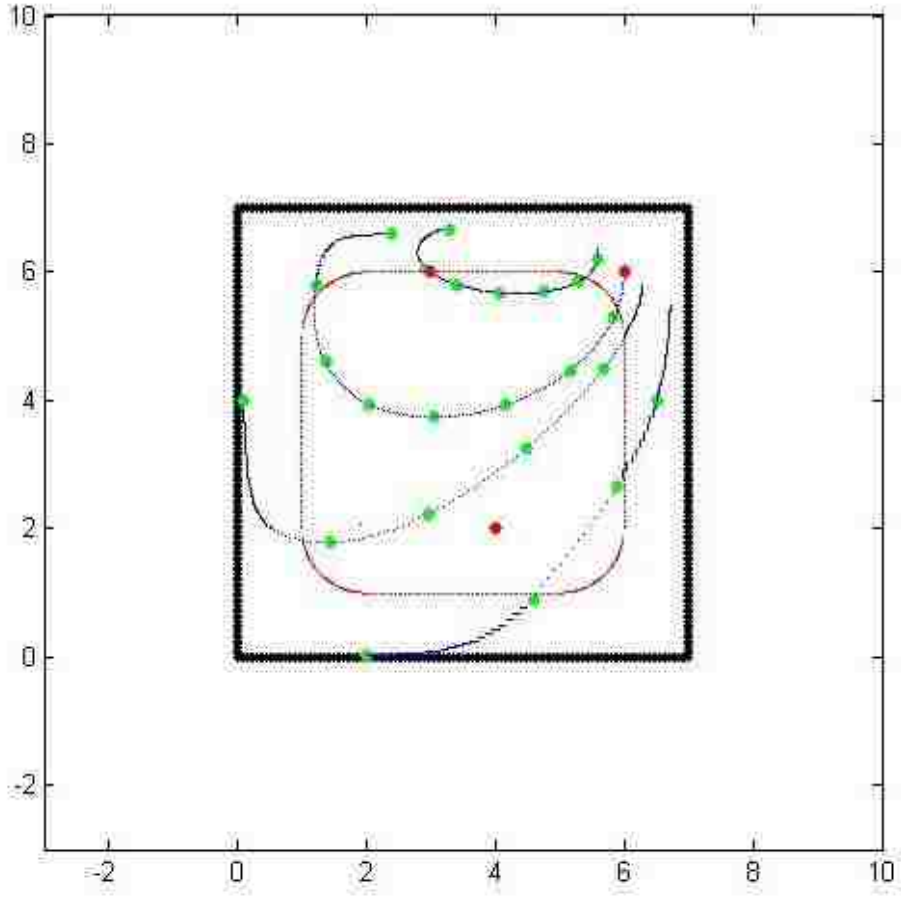


Figure 2.14. Boundary confined air-borne particle paths going through two sensors with twice the air speed.

2.4. THE SUMMARY OF PLOTTING AND STOPPING RULES

Step 1: confirm the order of the sensors

There are several sensors in the room where the author want to detect the source, the author should got the correct order from the all permutations. For excluding the incorrect and contradictory order, the author have set the rule that the author think the air flow can propagate along the shorter route. The process is showed below:

1. Using the cubic parameter interpolation method, the author can get the curve between any two sensors.

2. Calculate the length of the above-mentioned curve, such as L_{AB}.

3. The author can get the length between the two sensors, when the sensors' order is flipped, such as L_{BA}.

4. Compare the lengths when the orders of the sensors are different. The author think the shorter one is correct order. Such as If $L_{AB} < L_{BA}$, the correct order is A to B.

5. Do a loop to get the correct order between each two sensors.

Step 2: interpolate the paths that connect the sensors

In our modeling, the author assume that there're only very few sensor data available, since it is difficult to place sensors and collect data in locations that aren't very accessible and user friendly. With a few sensors, the author collect the concentrations levels of the chemical particle, the wind direction, and the local dissipation of the particle. Using cubic Hermite parameter interpolation, the author can get the paths that connect the sensors. Moreover, the author can adjust the parameter to get the extrapolation.

The problem with Hermite interpolation is the need to specify the derivatives at the endpoints of each section of the curve. Suppose the curve has n+1 data points (x0,y0), ⋯, (xn, yn), and the author wish to parameterize the cubic to allow complex features. Then the author must specify $x'(t_i)$ and $y'(t_i)$ for each i= 0, 1, ⋯,n , where $(x_i, y_i) = (x(t_i), y(t_i))$. This is not as difficult as it would first appear, however, since each portion can be generated independently, provided that the author ensure that the derivatives at the endpoints of each portion match those in the adjacent portion. Essentially, then, the author can simplify the process to one of determining a pair of cubic Hermite polynomials in the parameter t, where $t_0 = 0$ and $t_1 = 1$, given the endpoint data (x(0), x(1), y(0), y(1)) and the derivatives dy/dx(at t= 0) and dy/dx(at t= 1). The natural form for determining x(t) and y(t) required the author specify $x'(0), x'(1), y'(0)\, and\, y'(1)$.

The explicit Hermite curve in x and y required specifying only the equations:

$$\frac{dy}{dx}(t=0) = \frac{x'(0)}{y'(0)}, \quad \frac{dy}{dx}(t=1) = \frac{x'(1)}{y'(1)}.$$

By multiplying $x'(0)$ and $y'(0)$ by a common scaling factor, the tangent line to the curve at (x(0), y(0)) remains the same , but the shape of the curve varies.

To further simplify the process, the derivative at an endpoint is specified graphically by describing a second point, called a guide point, on the desired tangent line. The farther the guide point is from the node, the more closely the curve approximates the tangent line near the node.

The node occur at (x0,y0) and (x1,y1), the guide point for (x0,y0) is (x0+ α_0 ,y0+ β_0), and the guide point for (x1,y1) is (x1- α_1 ,y1- β_1). The cubic Hermite polynomial x(t) on [t0, t1] must satisfy

X(0)= x0, x(1)= x1, x'(0)= α_0 , x'(1)= α_1 .

It is easily verified that the unique cubic polynomial satisfying these conditions is

$$x(t) = [2(x_0 - x_1) + (\alpha_0 + \alpha_1)]t^3 + [3(x_1 - x_0) - (\alpha_1 + 2\alpha_0)]t^2 + \alpha_0 t + x_0.$$

In the similar manner, the unique cubic polynomial for y is

$$y(t) = [2(y_0 - y_1) + (\beta_0 + \beta_1)]t^3 + [3(y_1 - y_0) - (\beta_1 + 2\beta_0)]t^2 + \beta_0 t + y_0.$$

Popular graphics programs use this type of system for their freehand graphic representations but in a slightly modified form. The Hermite cubics are described as Bezier polynomial, which incorporate a scaling factor of 3 when computing the derivatives at the endpoints. This modifies the parametric equation to

$$x(t) = [2(x_0 - x_1) + 3(\alpha_0 + \alpha_1)]t^3 + [3(x_1 - x_0) - 3(\alpha_1 + 2\alpha_0)]t^2 + 3\alpha_0 t + x_0$$

In the similar manner, the unique cubic polynomial for y is

$$y(t) = [2(y_0 - y_1) + 3(\beta_0 + \beta_1)]t^3 + [3(y_1 - y_0) - 3(\beta_1 + 2\beta_0)]t^2 + 3\beta_0 t + y_0$$

We can construct a set of cubic Bezier curves (C0, C1,…， Cn-1) based on the parametric equations.

Where Ci is represented by

$$x_i(t), y_i(t) = (a_0^i + a_1^i t + a_2^i t^2 + a_3^i t^3, b_0^i + b_1^i t + b_2^i t^2 + b_3^i t^3).$$

Step 3: some rules to remove some unreasonable points and their links.

Assuming the each two points can have a link, the author can get countless new points and links. So, the author should set up some rules to remove some unreasonable points and their links.

a. Check order

All the links should obey the region order. In our case, all links should follow the order (region 1 to region 3, region 1 to region 2 to region 3)

b. Check process

We can get only one link derived from one link. It's not allowed that there are two different links from one point.

c. Check association

Firstly, the author give the definition of point association. If one point is generated by another point along another one's perpendicular line, the author think there are belonged to a same group. The author think there are not link between each two points in the same group.

d. Check group

If the link with the other existed one have same endpoints and beginning points that are belonged to the same group. The author think the new link should be removed.

e. Check original points (sensors)

Every muti-links should go through one point that is the location of any one sensor.

We should check each above rule. If one of rules isn't obeyed, the link should be removed. Finally, the author find if there are n sensors, the author will have n*(n-1) links and n multi-links.

Step 4: scaled correctly points on the path showing the speed of propagation

In the points where there are the sensors, the author can get the speed of the airflow from the sensor data. But, the author want to show the speed of every point in the whole room in our map.

In the step 2, the author have got the paths on which there are a few sensors. In this step, the author want to scale correctly points on the path showing the speed of propagation. Using the speed of the points where the sensors are located, the author can get the speed of propagation on every point. In our case, the author think the speed has the linear change along the propagation paths. The beginning point is the sensor point as the author know the speed in these points. The author draw the speeding points using the equal time interval. Through the speed value and the time interval, it is simple to get the

next speed point. Step by step, the author can finish the all speed point along the propagation direction on the path. In the map, the author can find the longer the distance between to speeding point indicate the speeding on this part is faster than the other one.

Step 5: map the whole room using interpolation and extrapolation

In the step 2, the author have got the paths on which there are a few sensors. But, the author want to topographically generate the flow diagram on a map of the surroundings and determine the probable source location of the chemical particles. There are two different cases when the author map the whole room. One is the region between two paths, the other is the region between a path and the boundary. When the author meet the first case, the author can use the method of bisection. Using two points from two different paths the author can get the midpoint. The author apply this method in a loop, the author can compute the whole path. As far as the other region, the author use a different method that based on the former method. The author calculate the normal vector of each points with a path that is belonged to the former region in order to get the crossing point. Then, the author generate a new point in the opposite direction. The distance from initial point and crossing point is proportional with the distance from initial point to the new point.

We're able to improve on the accuracy of the route as well as generate more viable routes. With the added accuracy and flexibility, we're also able to eliminate unfeasible options and conflicting sources.

Because every path is consisted of many discrete points, the author can using dichotomization to get several new points. Then, the new points together form a new path. Finally, the author can map the whole room in order to mark the chemical situation everywhere.

Step 6: considering boundary condition to modify and stop the airflow path

Several difficulties arise when formulating a shape transformation. The instance in which boundary information signals the presence of a shape is not clear. The decision of which boundary information to include and exclude in a description of shape is also nontrivial. This decision process poses both a local and a global problem. Over a localized region of a shape, the transformation should separately shape boundary information elements related to different shapes, even if shapes extend over a very wide

spatial region. On a more global scale the transformation must be able to integrate local shape encodings as subparts of the same global structure.

It's known that the boundary of room can affect the airflow's propagation. If there is not the boundary, the path would pass it. But, in fact, the propagating paths are affected by the boundary, they will spread along the boundary. The airflow near the boundary cannot swerve suddenly, so the author can assume an area that is very near the boundary, if the airflow enters into the area, it will be changed by the boundary. If the airflow is out off this area, it will not be affect by the boundary.

The method using proportional squeeze is used for the obstacles. If there is not the boundary existed, the airflow will go beyond it. the author find using tanh function to squeeze the distance between boundary and airflow can make the airflow approximate the real airflow path. Firstly, the author calculate the normal vector of each points that are beyond the inside boundary, and the normal will have a crossing point with the inside boundary. Secondly, the author can calculate the distance between the point and the crossing point, then the author make the distance shorter using tanh function. Finally, the author can get a new point that is in the boundary area. The author find the tanh can make our path satisfy this really condition that the closer the point with the inside boundary, it will get more effect by the boundary. When the point is near the inside boundary, the path just have a tiny change with the path without considering the boundary. When the points is far from the boundary, the path will be squeezed sharply. The author apply this method in a loop, the author can compute the whole path.

When the airflow curve is in the boundary region, the blocking stop the airflow curve when it is parallel with the boundary.

The error cause by the odor values, and the error between the assumed odor values is bounded.

A． assume ds/dx is perfect

We assume the propagation is totally correct, and the sensed values at two sensor's locations are S1 and S2.

The decay ratio along the propagation can be expressed by the $S = Ae^{-al}$, so the values on the perpendicular direction should be $s_1 e^{-al_1}$ and $s_2 e^{al_2}$.

We assume the sensed value's function on the perpendicular direction should be $S = ml + n$, then in our case $S = m_1 l_3 + n_1$ and $S = m_2 l_4 + n_2$.

Plugging $s_1 e^{-al_1}$ and $s_2 e^{al_2}$ into $S = m_1 l + n_1$ and $S = m_2 l + n_2$,

We get S= $m_1 l_3 + s_1 e^{-al_1}$ and $m_2 l_4 + s_2 e^{al_2}$. Because the author assume the ds/dx is perfect, the updated path should be identical with the original, which mean the S= $m_1 l_3 + s_1 e^{-al_1}$ and $m_2 l_4 + s_2 e^{al_2}$ should be equal.

If the sensed value's function on the perpendicular direction should be $S = ml + n$ cannot satisfy the physical propagation, the errors exist between the assumed odor values. The bound of the error is $[0, \left| m_1 l_3 + s_1 e^{-al_1} - m_2 l_4 - s_2 e^{al_2} \right|]$. Obviously, the author can easily prove the region is bounded.

The error the author get is based on the derivative of sensed values (ds/dl) and the sensed values (s).

B. assume the derivative of sensed values (ds/dl) has some percentage off.

We assume the correct sensed value's function on the perpendicular direction is $S = m_d l + n$, so the desired parameter and designed parameter's deviation is $\alpha = \frac{m_d - m}{m_d}\%$.

So, the bound of error $m_1 l_3 + s_1 e^{-al_1} - m_2 l_4 - s_2 e^{al_2}$ should be based on $\alpha = \frac{m_d - m}{m_d}\%$, the maximum error can be derived,

$$m_1 l_3 + s_1 e^{-al_1} - m_2 l_4 - s_2 e^{al_2} - m_{1d} l_3 - s_1 e^{-al_1} + m_{2d} l_4 + s_2 e^{al_2}$$

$$= (m_1 - m_{1d}) l_3 - (\mathrm{m}_2 - m_{2d})\mathrm{l}_4 .$$

Moreover, the author can get the error percentage is $\frac{2(m_1 - m_{1d}) l_3 - 2(\mathrm{m}_2 - m_{2d})\mathrm{l}_4}{m_{1d} l_3 + s_1 e^{-al_1} m_2 l_4 + s_2 e^{al_2}}$ %, and the percentage is based on the $\alpha = \frac{m_d - m}{m_d}\%$.

2.5. USING MATLAB TO STIMULATE THE PARTICLE PATHS

1. Identify the number of different s values.

Endpoints are at (x0, y0) and (x1, y1). The guide point for (x0, y0) is (x0+ α_0 , y0+ β_0), and the guide point for (x1, y1) is (x1- α_1 , y1- β_1).

2. Find new vectors array for each s value.

3. Do the interpolation.

This parametric equation for x is

$$x(t) = [2(x_0 - x_1) + 3(\alpha_0 + \alpha_1)]t^3 + [3(x_1 - x_0) - 3(\alpha_1 + 2\alpha_0)]t^2 + 3\alpha_0 t + x_0 .$$

In the similar manner, the unique cubic polynomial for y is

$$y(t) = [2(y_0 - y_1) + 3(\beta_0 + \beta_1)]t^3 + [3(y_1 - y_0) - 3(\beta_1 + 2\beta_0)]t^2 + 3\beta_0 t + y_0 .$$

4. Plot the same s value trajectories.

We can construct a set of cubic Bezier curves (C0, C1,…， Cn-1) based on the above parametric equations, where Ci is represented by

$$x_i(t), y_i(t) = (a_0^i + a_1^i t + a_2^i t^2 + a_3^i t^3, b_0^i + b_1^i t + b_2^i t^2 + b_3^i t^3).$$

Strep 1;

For each i=0, 1, …， n-1 do step 2 and 3.

Step 2;

Set $a_0^i = x_i$;

$b_0^i = y_i$;

$a_1^i = 3\alpha_0$;

$b_1^i = 3\beta_0$;

$a_2^i = 3(x_1 - x_0) - 3(\alpha_1 + 2\alpha_0)$;

$b_2^i = 3(y_1 - y_0) - 3(\beta_1 + 2\beta_0)$;

$a_3^i = 2(x_0 - x_1) + 3(\alpha_0 + 2\alpha_1)$;

$b_3^i = 2(y_0 - y_1) + 3(\beta_0 + 2\beta_1)$;

Step 3;

Output the all above coefficient.

The Matlab Script is attached below.

```
clear all
close all
clc
n=3; % n is the # of sensors
x=[3,2,6];
y=[6,2,6];  % give the value of the endpoint(x,y)
xjia = [3.3,2.8];
yjia =[6.5,3.0];    % give the value of the left guide point
xjian = [0,2.8,2.5];
yjian =[0,3.0,2.8];  % give the value of the left guide point
a0 = [0,0];
a1 = [0,0];
a2 = [0,0];
a3 = [0,0];
b0 = [0,0];
b1 = [0,0];
b2 = [0,0];
b3 = [0,0];   % initiate  the coefficient of the curve
%generate the coefficient of the curve
for i=1:n-1
    a0(i)=x(i);
    b0(i)=y(i);
    a1(i)=3*(xjia(i)-x(i));
    b1(i)=3*(yjia(i)-y(i));
    a2(i)=3*(xjia(i)+xjian(i+1)-2*x(i));
    b2(i)=3*(yjia(i)+yjian(i+1)-2*y(i));
    a3(i)=x(i+1)-x(i)+3*xjia(i)-3*xjian(i+1);
    b3(i)=y(i+1)-y(i)+3*yjia(i)-3*yjian(i+1);
end
% generate the cubic bezier curves Co, C1 in parametric
for i=1:n-1;
    p1=[a3(i),a2(i),a1(i),a0(i)];
    xt=poly2sym(p1,'t');
    p2=[b3(i),b2(i),b1(i),b0(i)];
    yt=poly2sym(p2,'t');
end
figure(1)
plot(x,y,[0,1]);  % plot the smell trajectories
```

3. COMPARE AND VALIDATE INTERPOLATION AND EXTRAPOLATION APPROACH

3.1. USING COMPUTATIONAL FLUID DYNAMICS TO GET THE ANALYTICAL SOLUTION

To compare and validate our approach, the author use exact analytical methods for simpler cases and use finite-element method based software (such as COMSOL) for more complicated cases.

The analysis of airborne particle motion is identical to the fluid motion analysis in physics. The fluid motion is governed by the Navier-Stokes nonlinear partial differential equation [25], such that motion in the two dimensional space satisfies:

$$\begin{aligned} &\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}=0, \\ &u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}=-\frac{1}{\rho}\frac{\partial P}{\partial x}+\nu(\frac{\partial^2 u}{\partial x^2}+\frac{\partial^2 u}{\partial y^2}), \\ &u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}=-\frac{1}{\rho}\frac{\partial P}{\partial y}+\nu(\frac{\partial^2 v}{\partial x^2}+\frac{\partial^2 v}{\partial y^2}), \end{aligned} \tag{1}$$

Where u and v are the components of the velocity in the x and y directions, ρ is the fluid density, and P is the pressure.

The analytical solutions to the Navier-Stokes equations depend on the initial and the boundary conditions, and the exact solutions exist only for simple cases.

In this section, the author will assume that the particle flow dynamics is two dimensional, and it's uncompressible, inviscid, and irrotational. If D is a simply connected domain in (x, y) and the flow is irrotational, the integral $\int udx+vdy$ is independent of path in D. If the author integrate from a fixed point (a,b) to a variable point (m,n), then the integral becomes a functions of the point (m,n):

$$\Phi(m,n)=\int_{(a,b)}^{(m,n)}(udx+vdy) \tag{2}$$

The function $\Phi(x, y)$ is called velocity potential of the motion. Since the integral is independent of path, $udx + vdy$ is an exact differential, namely, the differential of function $\Phi(x, y)$, that is,

$$udx + vdy = \frac{\partial \Phi}{\partial x} dx + \frac{\partial \Phi}{\partial y} dy \tag{3}$$

Form (3),

$$u = \frac{\partial \Phi}{\partial x}, \quad v = \frac{\partial \Phi}{\partial y} \tag{4}$$

By substituting u and v in (4), into (1) the author see that $\Phi(x, y)$ satisfies Laplace's equation:

$$\nabla^2 \Phi = \frac{\partial^2 \Phi}{\partial x^2} + \frac{\partial^2 \Phi}{\partial y^2} = 0 \tag{5}$$

By the above theoretical derivation, the author use Laplace's equation to model fluid motion.

Let $\Psi(x, y)$ be a conjugate function of $\Phi(x, y)$ .The function $\Psi(x, y)$ is called the stream function of the flow. The curves $\Psi(x, y) = const$ are the streamlines of the fluid. The author know that both $\Psi(x, y)$ and $\Phi(x, y)$ have continuous second derivatives. Then the complex function

$$F(x, y) = \Phi(x, y) + i\Psi(x, y) \tag{6}$$

is analytic in the region of the flow. This function is called the complex potential of the flow.

The velocity of the flow can be obtained by differentiating (6) and using Cauchy-Riemann equations; the author find

$$F_x(x, y) = \frac{\partial \Phi}{\partial x} + i\frac{\partial \Psi}{\partial x} = \frac{\partial \Phi}{\partial x} - i\frac{\partial \Phi}{\partial y} = u - iv \tag{7}$$

General solution to (7) can be complicated and unnecessary for the simple cases that the author are considering. Indeed, the author will assume the function F from the initial flow and determine the specifics by substituting the functions into the differential equations. Because of the uniqueness of the solution to the differential equation under

initial and boundary conditions, if the function F satisfies (5) and the boundary conditions, then it's the unique solution.

Case 1: fluid flow with free boundary.

As the first case, the author choose the no boundary case with a uniform infinite width flow of particles at a certain angle α , as shown in Figure 3.1. Since the solution of the partial differential equation is unique, the author can firstly assume a solution and then check it satisfies or not. If the author represent $F(\mathrm{x},\mathrm{y}) = \mathrm{kz} = k_1 x + ik_2 y$ (K real number) describe a uniform flow in the $x - y$ plane.

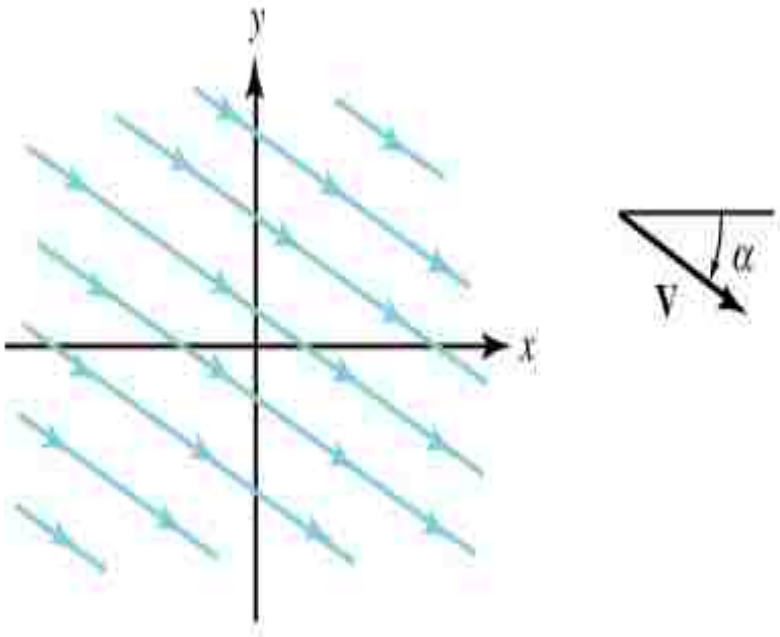


Figure 3.1. A uniform parallel flow.

If the flow is irrotational then $\nabla \times F = 0$ is automatically satisfied by writing $F = -\nabla\phi$, where ϕ is termed the velocity potential, hence $F_y = -\frac{\partial \psi}{\partial y}$ and $F_y = -\frac{\partial \phi}{\partial y}$. On the other hand, if the flow is incompressible then $F_x = -\frac{\partial \psi}{\partial y}$ and $F_y = \frac{\partial \psi}{\partial x}$, where ψ is termed the stream function.

Therefore, substituting $F(\mathrm{x}, \mathrm{y}) = \mathrm{kz} = k_1 x + ik_2 y$ into the differential equations, the author can get the velocity potential is $\Phi(x, y) = k_1 x - k_2 y$ and the stream function is $\psi(x, y) = k_2 x + k_1 y$, where the author can find the streamlines and potential lines are orthogonal.

The velocity of the flow can be obtained:

$$V(\mathrm{x}, \mathrm{y}) = \overline{\mathrm{F}'(\mathrm{x}, \mathrm{y})} = \overline{k_1 x + ik_2 y} == k_1 x - ik_2 y$$

The streamlines are parallel lines given by the equation $k_2 x + k_1 y = \text{constant}$ and are inclined at an angle $\alpha = -\arctan(\frac{k_2}{k_1})$.

Picking arbitrary two points in the $x - y$ plane ((x_1, y_1) and (x_2, y_2)) be the two sensor's locations, the author use the interpolation method can get the fluid propagation path between the two points. However, after considering the chemical concentration, the endpoints of the path should be modified. When the two points are chosen on a same streamline, the streamline can be calculated to $k_2 x + k_1 y = \text{constant}$. The two methods get the same path, so that the error term is equal to zero.

The below Figure 3.2 gives the error curve between fluid dynamic method and our proposed method. The left one is fluid path plot using our proposed method, the right one is the error curve. For a certain path, the author select this red path be the considered path, then fix one point $x_1 = 1$, change x_2 in the region $(0, \infty)$. The error curve shows the error value of fluid path derived by our proposed method.

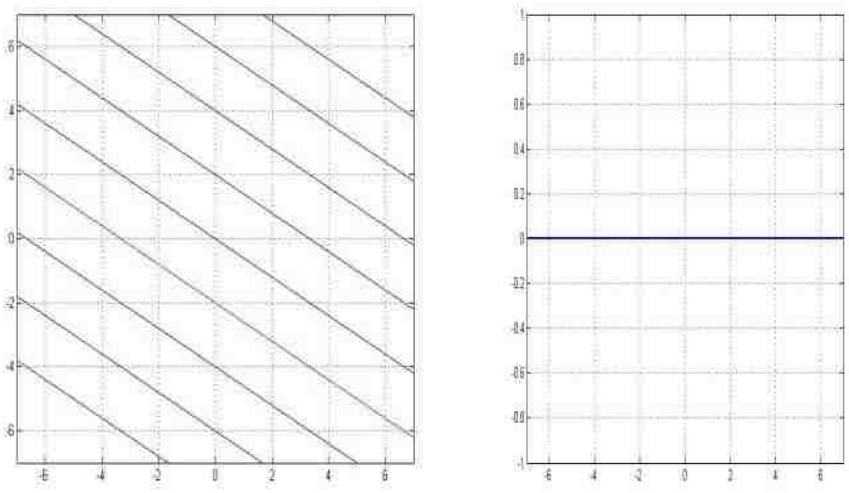


Figure 3.2. The performance of interpolation and extrapolation for a uniform parallel flow.

Case 2: fluid flow with infinite wall.

In this case, as shown in Figure 3.3, the complex potential is $F(\mathrm{x},\mathrm{y}) = \frac{A}{2} z^2 = \frac{A}{2}(x^2 - y^2) + iAxy$, where A is a positive real number. The velocity potential and stream function are given by $\Phi(x, y) = \frac{A}{2}(x^2 - y^2), \psi(x, y) = Axy$.

The streamlines $\psi(x, y) = \text{constant}$ are from a family of hyperbolas with asymptotes along the coordinate axes. The velocity vector $V(\mathrm{x},\mathrm{y}) = \overline{\mathrm{F}'(\mathrm{x},\mathrm{y})} = A(\mathrm{k}_1 x - k_2 y)$ indicates that in the upper half-plane $\mathrm{Im(z)} > 0$, the fluid flows down along the streamlines and spreads out along the x axis, as against a wall.

Picking arbitrary two points in the $x - y$ plane ((x_1, y_1) and (x_2, y_2)) be the two sensor's locations, the author use the interpolation method can get the fluid propagation path between the two points. However, after considering the chemical concentration, the endpoints of the path should be modified. When the two points are chosen on a same streamline, the streamline can be calculated to $y = ax^3 + bx^2 + cx + d$.

The error term is $\int_{(x_1,y_1)}^{(x_2,y_2)}(\frac{k}{x}-ax^3-bx^2-cx-d)^2dx$, where the K is a real number and (a,b,c,d) are four parameters that is related with the coordinates of the two points and derivatives of the two points . In this case, substituting the (a,b,c,d), the author get the error term is

$$E = k^2(\frac{1}{5}(x_2^5-x_1^5)-\frac{(x_1+x_2)}{2}(x_2^4-x_1^4)+\frac{1}{3}(x_1^2+x_2^2+4x_1x_2)(x_2^3-x_1^3) \\ -x_1x_2(x_1+x_2)(x_2^2-x_1^2)+x_1^2x_2^2(x_2-x_1)$$

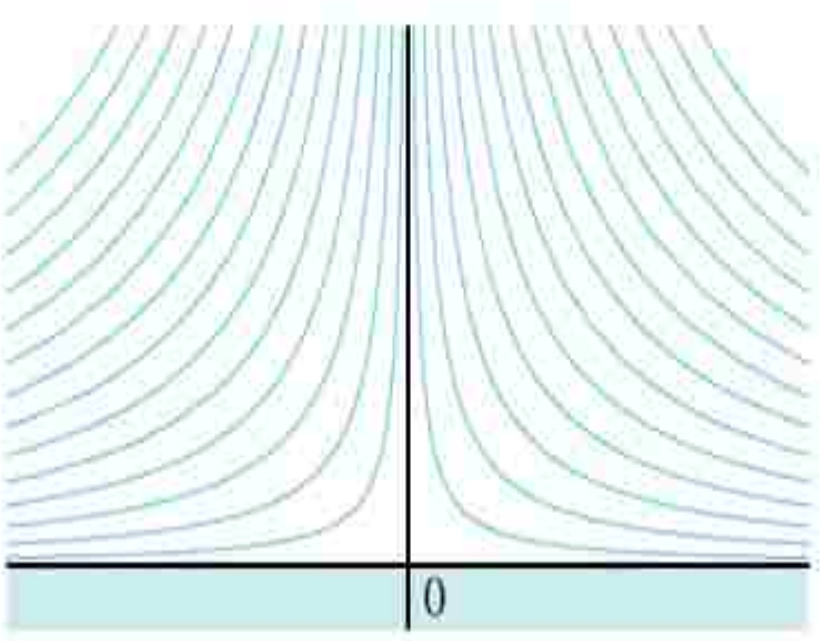


Figure 3.3. The fluid flow around infinite wall.

The below figure give the error curve between fluid dynamic method and our proposed method. The left one is fluid path plot using our proposed method, the right one

is the error curve. For a certain path, the author select this red path be the considered path, then fix one point $x_1 = 1$, change x_2 in the region $(0, \infty)$. In the Figure 3.4, the error curve shows the error value of fluid path derived by our proposed method.

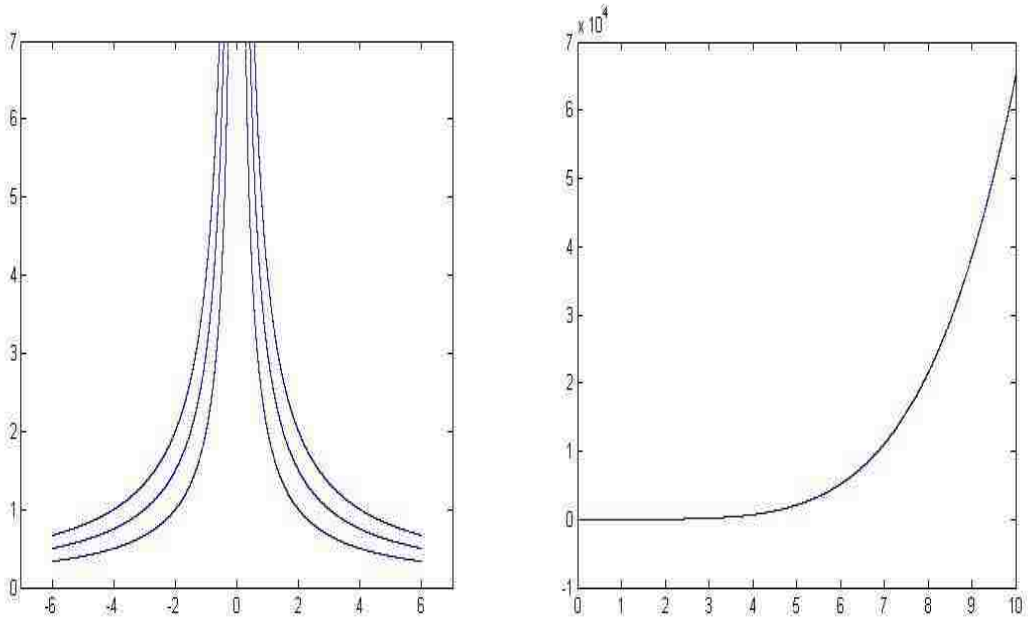


Figure 3.4. The performance of interpolation and extrapolation for the fluid flow around infinite wall.

Case 3: inviscid flow past a cylindrical obstacle**.**

In this case, as shown in Figure 3.5, the complex potential for an ideal fluid flowing from left to right across the complex plane and around the unit circle $|z| = x^2 + y^2 = 1$.

We use the fact that the conformal mapping $w = S(z) = z + \frac{1}{z}$ maps the domain $D = \{z : |z| < 1\}$ one-to-one and onto the w plane slit along the segment $-2 \le u \le 2, v = 0$. The complex potential for a uniform horizontal flow parallel to this slit in the w plane is $F_1(w) = Aw$, where A is a positive real number. The stream function for the flow in the w plane is $\psi(u, v) = Av$ so that the slit lies along the streamline $\psi(x, y) = 0$.

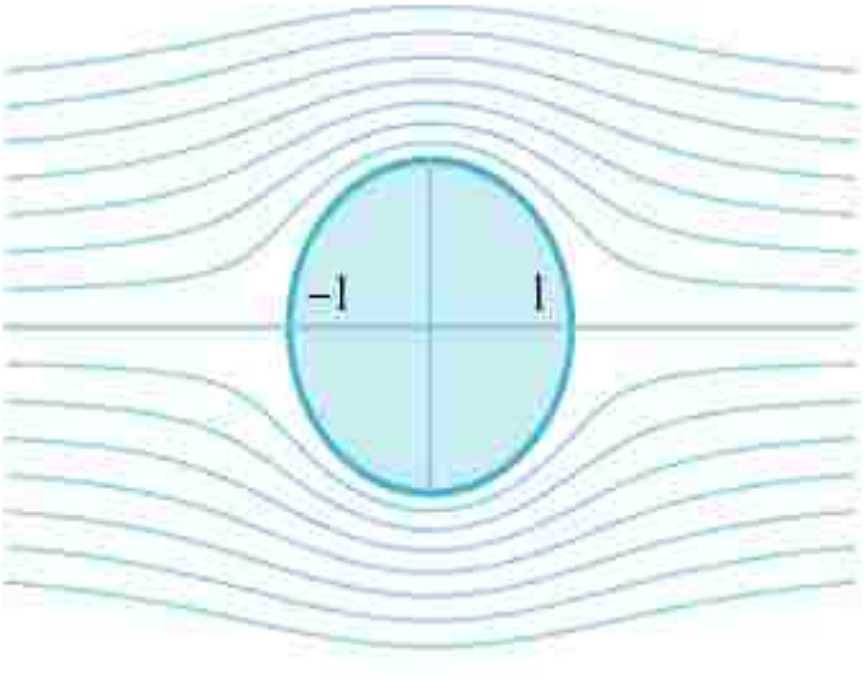


Figure 3.5. Fluid flow around a circle.

The composite function $F_2(z) = F_1(\mathrm{S(z)})$ determines the fluid flow in the domain D, where the complex potential is $F_2(z) = F_1(\mathrm{S(z)}) = \mathrm{A(z+}\frac{1}{z}\mathrm{)}$, where $A > 0$. the author can use polar coordinates to express $F_2(z)$ as

$$\begin{aligned} F_2(z) &= F_2(re^{i\theta}) \\ &= \mathrm{A}(re^{i\theta} + \frac{1}{re^{i\theta}}) \\ &= A(\mathrm{r}\cos\theta + ir\sin\theta + \frac{1}{r\cos\theta + ir\sin\theta}) \\ &= A(\mathrm{r}\cos\theta + ir\sin\theta + \frac{r(\cos\theta - ir\sin\theta)}{r^2}) \\ &= \mathrm{A(r+}\frac{1}{r}\mathrm{)}\cos\theta + iA\mathrm{(r-}\frac{1}{r}\mathrm{)}\sin\theta \end{aligned}$$

The streamline $\psi(r,\theta) = A(r - \frac{1}{r})\sin\theta = 0$ consists of the rays $r > 1, \theta = 0 \; and \; r > 1, \theta = \pi$ along the x axis and the curve $r - \frac{1}{r} = 0$, which is the unit circle $|z| = r = 1$. Thus the unit circle can be considered as a boundary curve for the fluid flow.

The approximation $F_2(z) = F_1(\mathrm{S(z)}) = \mathrm{A(z+}\frac{1}{z}\mathrm{)} \approx \mathrm{Az}$ is valid for large values of z, so the author can approximate the flow with a uniform horizontal flow having speed $|V(x,y)| = A$ at points that are distant from the origin.

Picking arbitrary two points in the $x - y$ plane ((x_1, y_1) and (x_2, y_2)) be the two sensor's locations, the author use the interpolation method can get the fluid propagation path between the two points. However, after considering the chemical concentration, the endpoints of the path should be modified. When the two points are chosen on a same streamline, the streamline can be calculated to $y = ax^3 + bx^2 + cx + d$. The error term is $\int_{(x_1,y_1)}^{(x_2,y_2)} (\frac{k}{Ax} - ax^3 - bx^2 - cx - d)^2 dx$, where the A is a positive real number and (a,b,c,d) are four parameters that is related with the coordinates of the two points and derivatives of the two points .

The below Figure 3.6 give the error curve between fluid dynamic method and our proposed method. The left one is fluid path plot using our proposed method, the right one is the error curve. For a certain path, the author select this red path be the considered path, then fix one point $x_1 = 1$, change x_2 in the region $(0, \infty)$. In the Figure 3.6, the error curve shows the error value of fluid path derived by our proposed method.

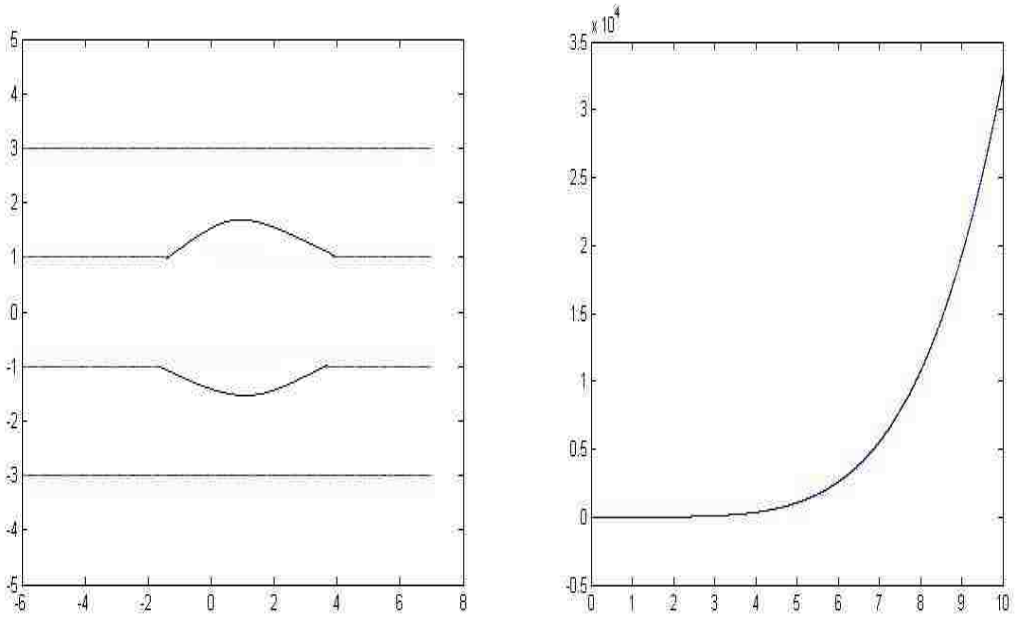


Figure 3.6. The performance of interpolation and extrapolation for the fluid flow around a circle.

In summary, based on all of the above results, they can conclude that the error term is related with the coordinates and derivatives of two sensors and the boundary condition. Obviously, it's coincident with the natural law. The closer distance of two sensors, the error term of the chemical propagation path is smaller. In the other words, the

more sensors the author use, the better result the author get, since the distance of any two sensors become closer. As for the boundary condition, the simplest condition is free boundary in which the error is equal to zero. It's because the cubic hermit Spline function can approximate any function, however, the path on the complex boundary will bring higher order error term. When the order of Spline function is infinite, the error of approximation should be zero.

3.2. LIPSCHITZ CONDITION

Theorem 4: Cubic spline function (x(t), y(t)) that is used to describe the particle paths satisfies Lipschitz continuity , such that,

$|x(t_1)-x(t_2)| \le L|t_1-t_2|$
$|y(t_1)-y(t_2)| \le L|t_1-t_2|$, there exists a constant $L>0$.

Proof:

Each segment of the cubic polynomial function is a cubic polynomial:

$p(t) = at^3+bt^2+ct+d = [t^3\ t^2\ t\ 1][a\ b\ c\ d]^T$

then, $p(t_1) = at_1^{\,3}+bt_1^{\,2}+ct_1+d$ and $p(t_2) = at_2^{\,3}+bt_2^{\,2}+ct_2+d$

$p(t_1)-p(t_2) = a(t_1^3-t_2^3)+b(t_1^2-t_2^2)+c(t_1-t_2)$

$$\left|\frac{p(t_1)-p(t_2)}{t_1-t_2}\right| = \left|a(t_1^2+t_2^2+t_1t_2)+b(t_1+t_2)+c\right|$$

Select $t_m = \max\{t_1,t_2\}$,

$$\left|\frac{p(t_1)-p(t_2)}{t_1-t_2}\right| \le \left|3at_m^2+2bt_m+c\right|,$$

Then, let $\left|3at_m^2+2bt_m+c\right| = L$, $\left|\frac{p(t_1)-p(t_2)}{t_1-t_2}\right| \le L$.

Theorem 5: Given M and M' to represent chemical concentration and its derivative, Let $M=(S,d)$ and $M'=(S',d')$ be metric spaces and $f: S \to S'$ be a mapping , then f is Lipschitz continuous mapping. Such that,

$\forall s_1, s_2 \in S : d'(f(s_1)-f(s_2)) \le Ld(s_1-s_2)$.

Proof:

Assume the $f: S \to S'$ is continuously differentiable function,

For every ε there exists $\delta(\varepsilon) > 0$ such that $|S_1 - S_2| < \delta$ imply $|f(s_1) - f(s_2)| < \varepsilon$.

Using Taylor's Theorem

$$f(s) = f(s_0) + \nabla f(ts + (1-t)s_0)^T (s - s_0),\ t \in [0,1]$$

Here $\nabla f(s)$ is the gradient of function.

$$f(s_1) - f(s_2) = \int_0^1 \nabla(ts_1 + (1-t)s_2)^T (s_1 - s_2) dt$$

$$\leq \|s_1 - s_2\| \int_0^1 \nabla(ts_1 + (1-t)s_2)^T dt$$

Using $L = \sup_z \|\nabla f(s)\|$, the above inequality can be

$$\forall s_1, s_2 \in S : d'(f(s_1) - f(s_2)) \leq L d(s_1 - s_2)$$.

4. EXPERIMENTAL RESULTS OF INDOOR AND OUTDOOR ENVIRONMENT

4.1. SIMULATION AND ANALYSIS IN AN EXPERIMENTAL ENVIRONMENT

In the above method, the author know the airflow is affected by the boundary. But, most of real cases have openings and obstacles. As a test case that exist obstacles and openings, the author consider a two sensor configuration system as in Figure 4.1 and a three sensor configuration system as in Figure 4.2.

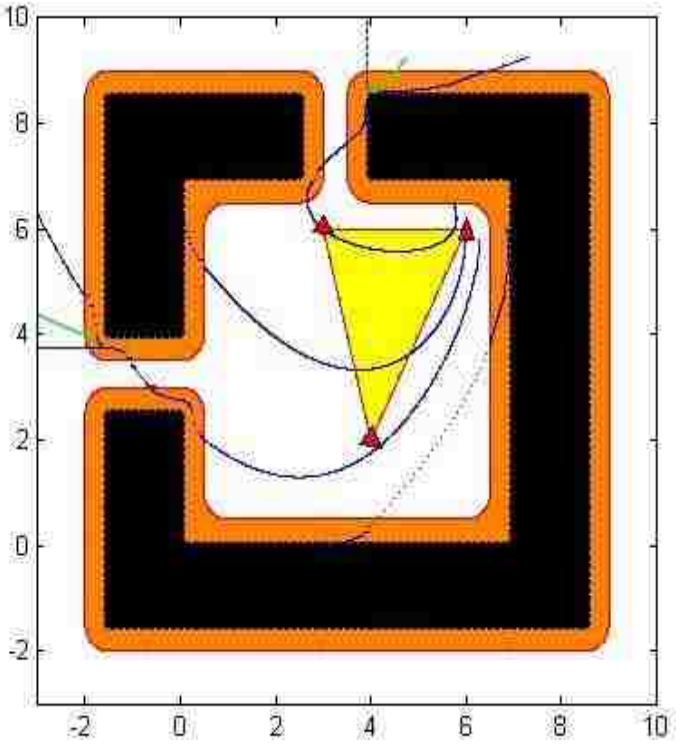


Figure 4.1. Boundary confined air-borne particle paths going through two sensors in a rectangular room with two openings.

The environment has the same basic information as follows: a search space with 10×10 m2, two air openings with width 1 m and an ethanol source of 100 ppm concentration. The coordinates of the two air inlets are (x = 0, y = [3, 4]) and (x = 7, y = [3, 4]), respectively.

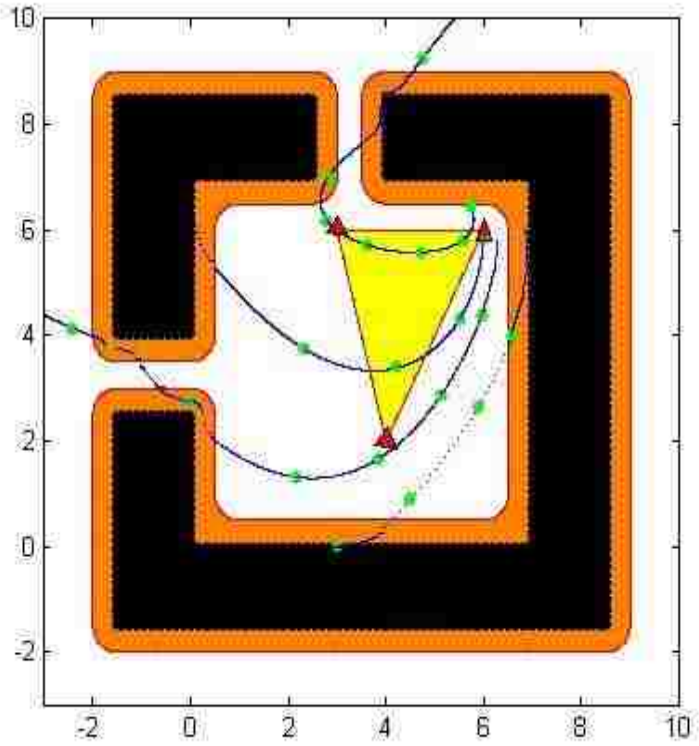


Figure 4.2. Boundary confined air-borne particle paths going through three sensors in a rectangular room with two openings.

In the figures, the black zones are walls and obstacles that block the chemical particle propagation. The orange zones are the neighborhood of the obstacles and walls.

The paths should be affected by any obstacles that may have been in the room. However, the change on the paths should be gradual in order to make them be the smooth curves.

At the same time, if the room has two opening on the walls, the paths will go through the opening rather than distorting the paths using the boundary condition. To solve the case with obstacles and openings, the author propose the following couple of assumptions. Figures 4.3 and 4.4 show a circle obstacle existed in the room.

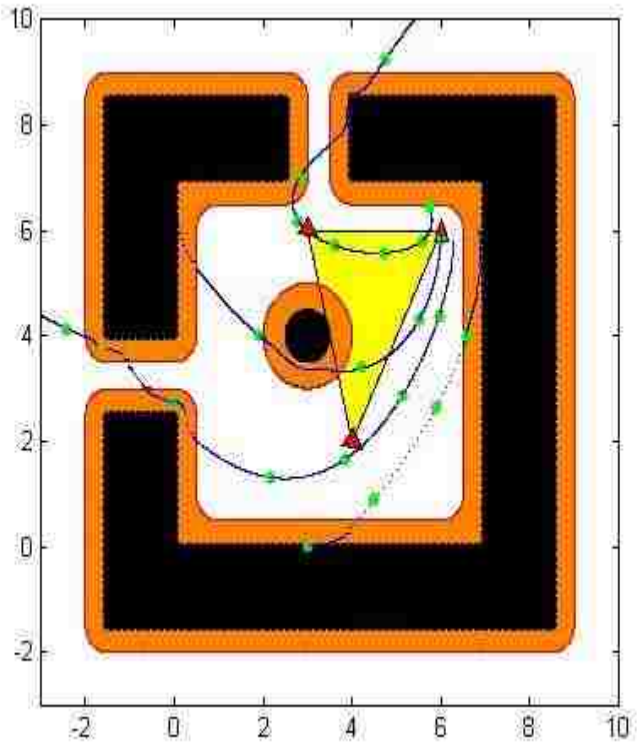


Figure 4.3. Air-borne particle paths going through two sensors in a rectangular room with a circle obstacle.

Assumption 6: The obstacles in the room make the path spread along their outline of obstacles.

Assumption 7: The chemical particle paths should depart the openings with a certain angle between the boundary and the original path.

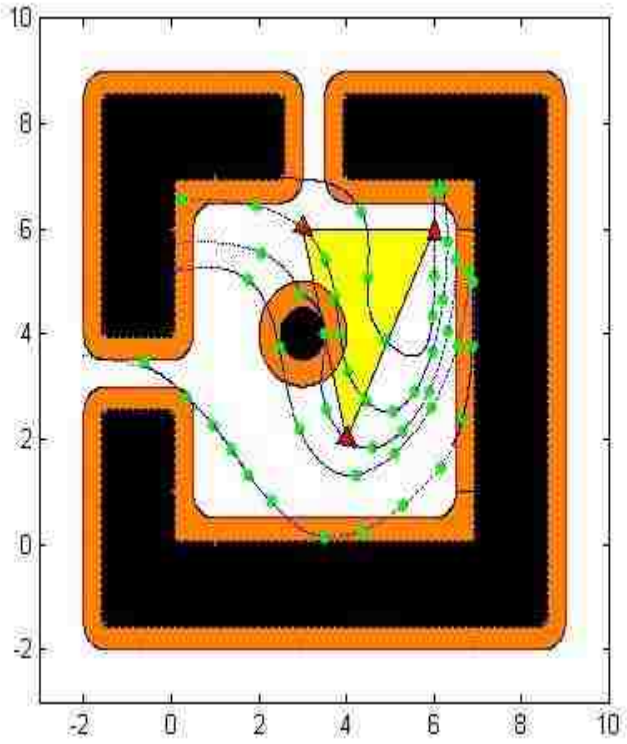


Figure 4.4. Air-borne particle paths going through three sensors in a rectangular room with a circle obstacle.

Base on the above assumptions, the chemical particle paths are distorted in the neighborhood of obstacles. Using above approach handling the obstacles, the author can map the paths inside the obstacles and the neighborhood of the obstacles onto the buffer

zone. The approach is similar with the one that used in handling boundaries. Hence, it enables the consistent location and derivative on the boundary of obstacles.

When the paths go through the opening, the portions outside the room boundary should neither follow the original path nor squeeze the path onto the buffer zone. If the author had applied the previous approach, the paths will be mapped onto the buffer zone. Obviously, it's unreasonable because the opening of wall cannot block paths. It's like a outlet for the room. The paths need depart the wall with a certain angle (0~45 degree) between the wall and the original path. The angle is depended on the velocity of the wind, density of the fluid and so on. In our case, this angle is set at 45 degree.

4.2. SIMULATION AND ANALYSIS ON AN REAL WORLD MAP

From the Google Earth, a real map of Missouri University of science and technology can be derived. Using the edge detection method, the processed map only exits the main buildings. Figure 4.5 showed the real map captured from Google Earth.

Edge detection includes a variety of mathematical methods that aim at identifying points in a digital image at which the image brightness changes sharply or, more formally, has discontinuities. The points at which image brightness changes sharply are typically organized into a set of curved line segments termed edges. The same problem of finding discontinuities in 1D signals is known as step and the problem of finding signal discontinuities over time is known as change detection. Edge detection is a fundamental tool in image processing, machine vision and computer vision, particularly in the areas of feature detection and feature extraction. Figure 4.6 shows the real map processed by the edge detection approach.

There are many methods for edge detection, as shown in Figure 4.6, but most of them can be grouped into two categories, search-based and zero-crossing based.

The search-based methods detect edges by first computing a measure of edge strength, usually a first-order derivative expression such as the gradient magnitude, and then searching for local directional maxima of the gradient magnitude using a computed estimate of the local orientation of the edge, usually the gradient direction. The zero-crossing based methods search for zero crossings in a second-order derivative expression computed from the image in order to find edges, usually the zero-crossings of the Laplacian or the zero-crossings of a non-linear differential expression. As a pre-

processing step to edge detection, a smoothing stage, typically Gaussian smoothing, is almost always applied (see also noise reduction).

Figure 4.5. A real map of Missouri University of Science and Technology from the Google Earth.

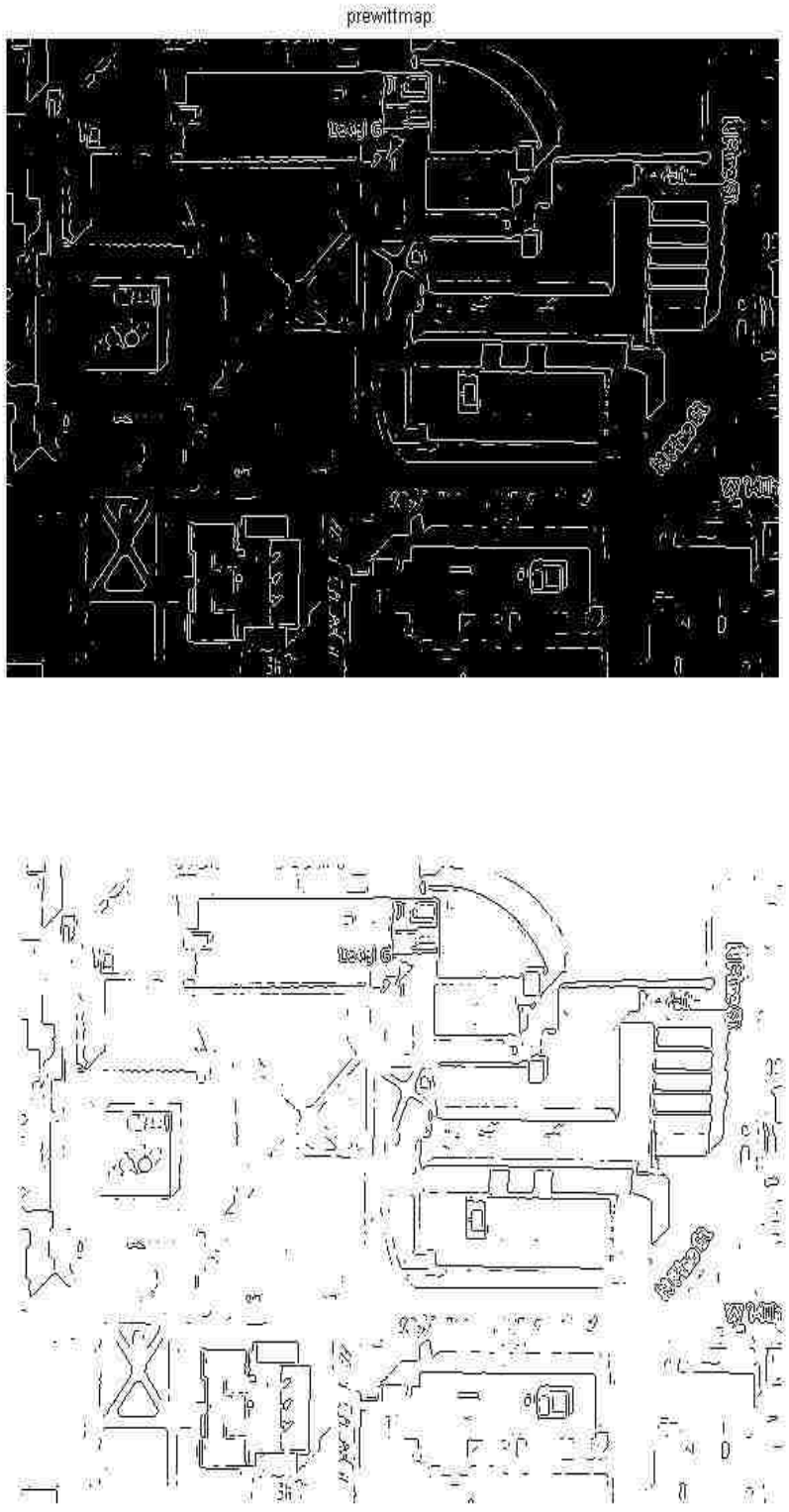


Figure 4.6. The campus map after different edge detection algorithms.

The edge detection methods that have been published mainly differ in the types of smoothing filters that are applied and the way the measures of edge strength are computed. As many edge detection methods rely on the computation of image gradients, they also differ in the types of filters used for computing gradient estimates in the x- and y-directions.

In the Figure 4.7, the map will be used to analyze the chemical particle propagation. After the edge detection, the feature has been extracted. The only buildings are left, because the buildings are the obstacles at which the chemical particle cannot be distributed.

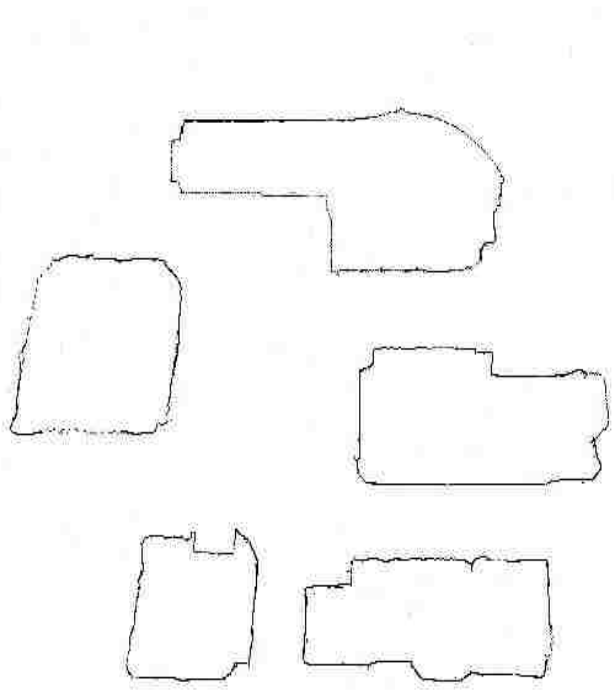


Figure 4.7. A real map of Missouri University of Science and Technology processed by edge detection method.

Using all of above approaches, the author can extend the path unlimitedly. However, in fact the entire paths are not completely exact. The author should know where the author can start a path and where the path ends to decide the effective range of the paths. The following assumption is introduced for the paths' credible section.

Assumption 8: Based on the measurement range of sensors, only the sections of the paths where are surrounded by sensors are credible.

Because the information collected by the sensors is local variable, only the data on the sensor location is completely correct and the entire paths are constructed based on the sensors' data.

The section of the paths where are near the sensors have more credible level, while the sections of the paths where are far from the sensors should have a low credible level. Using all sensors be the vertices, the author can plot a convex polygon in which the author believe the paths are completely correct. Keeping an equal distance with the polygon, the author get a zone which is encircled. In this encircled section, the paths are correct in certain probability.

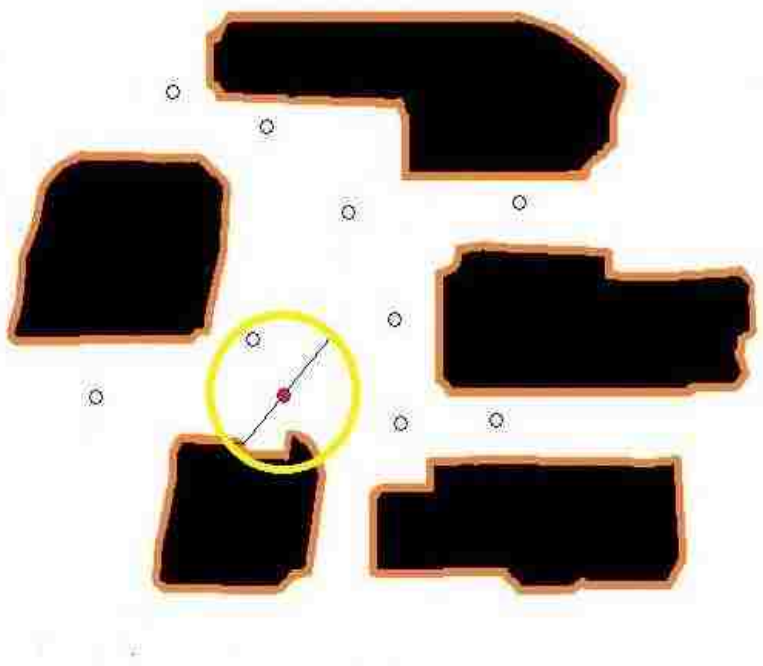
In the Figures 4.8, the green dotted lines show the exact zone, the yellow dotted lines designate the uncertain region. At the same time, the boundary lines of uncertain zone truncate the particle paths. Hence, the author are able to obtain both the starting points and ending points of the paths. Beyond this uncertain zone, the path is not thought be credible. It's because the portions beyond the uncertain zone are derived by extrapolation without any updated data from nearby sensors.

To verify the effectiveness of the proposed assumptions, real simulation is developed by COMSOL software that is used to analyze complex flow of fluid dynamics. COMSOL Multiphysics is a finite element analysis, solver and simulation software / FEA software package for various physics and engineering applications, especially coupled phenomena, or multi-physics.

The package is cross-platform (Windows, Mac, Linux). In addition to conventional physics-based user interfaces, COMSOL Multiphysics also allows entering coupled systems of partial differential equations (PDEs).

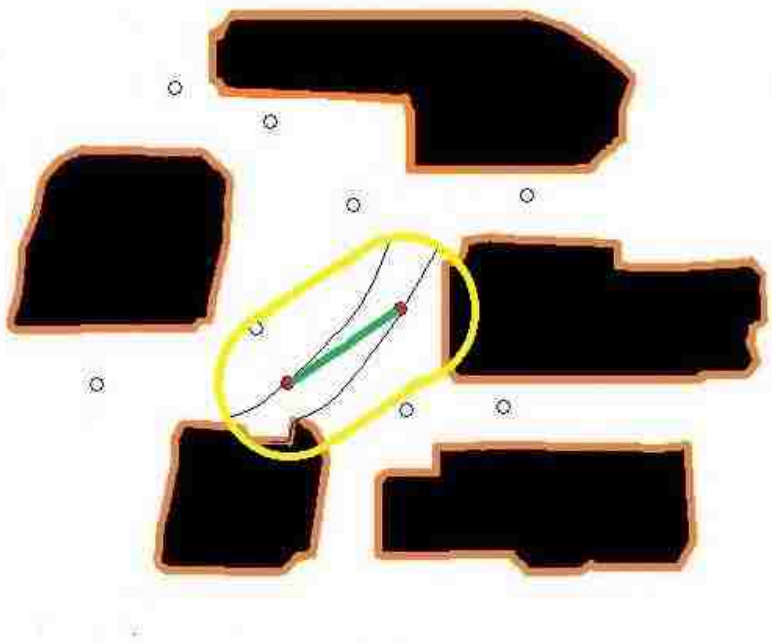
The PDEs can be entered directly or using the so-called weak form. Since version 5.0 (2014), COMSOL Multiphysics is also used for creating physics-based apps. These

apps can be run with a regular COMSOL Multiphysics license but also with a COMSOL Server license. An early version (before 2005) of COMSOL Multiphysics was called FEMLAB.

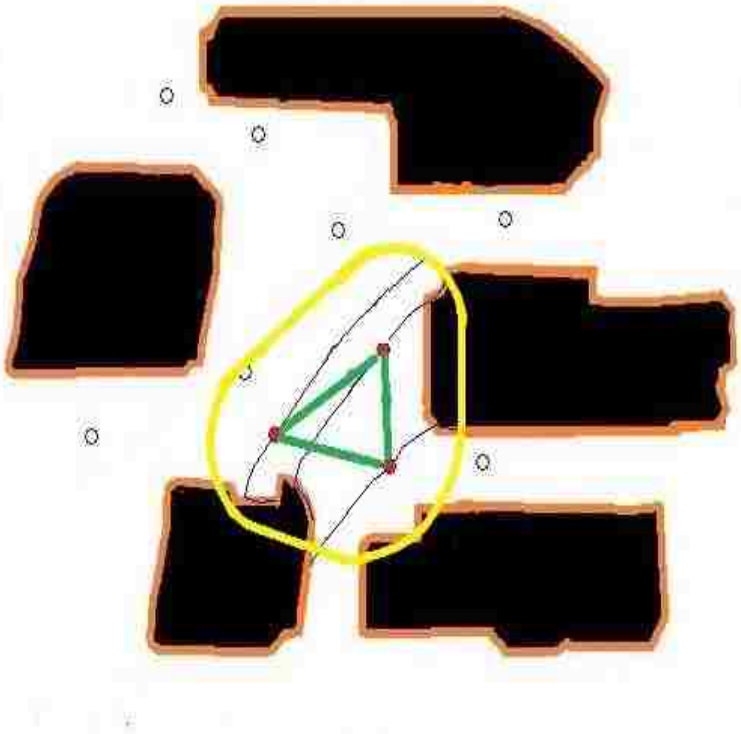


(a)

Figure 4.8. The credible sections of chemical path using different number of working sensor.

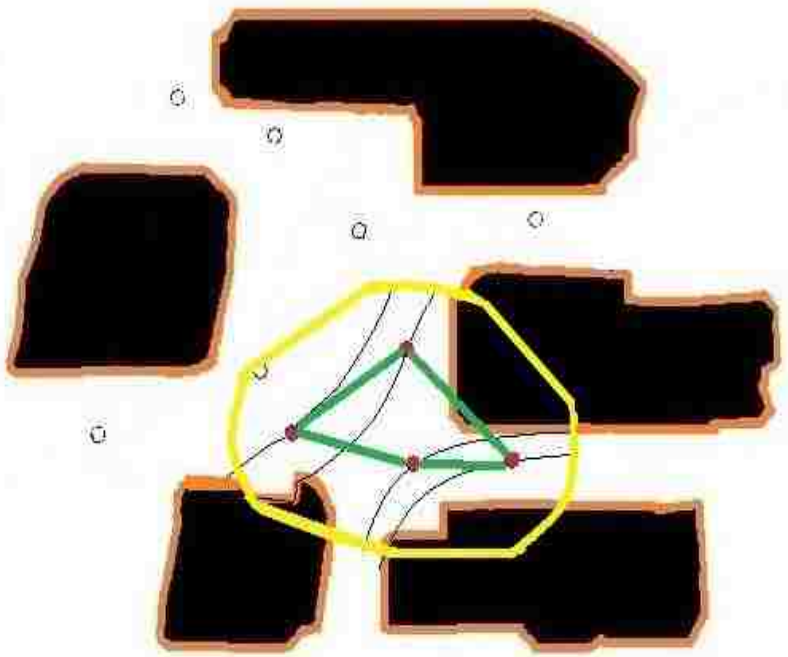


(b)

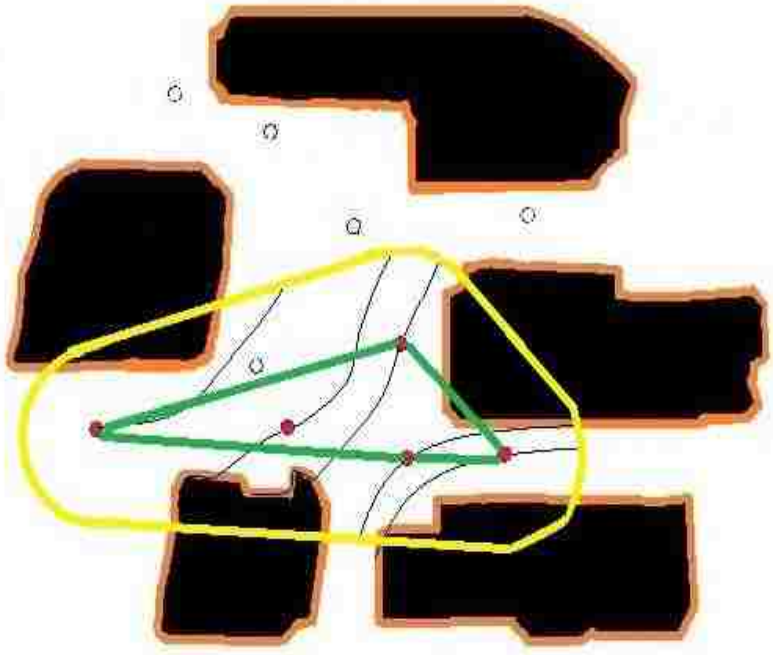


(c)

Figure 4.8. The credible sections of chemical path using different number of working sensor (cont.).

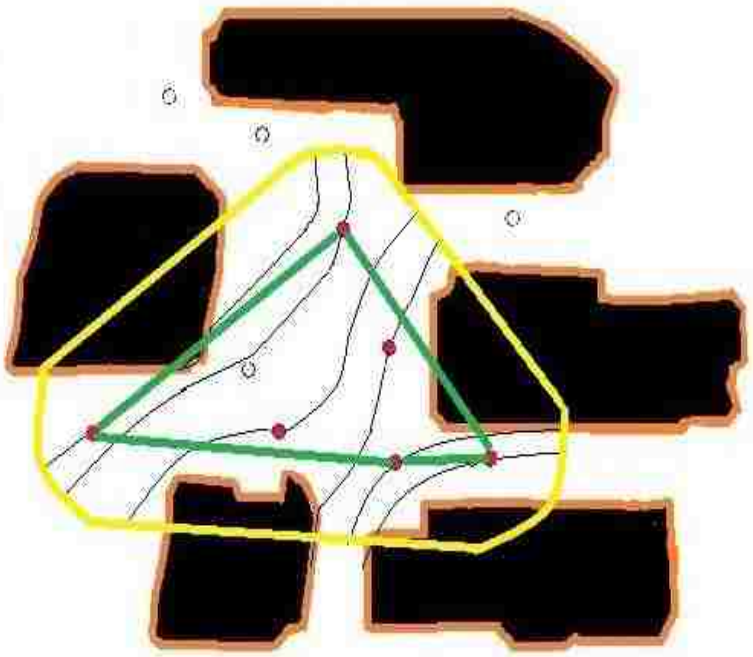


(d)

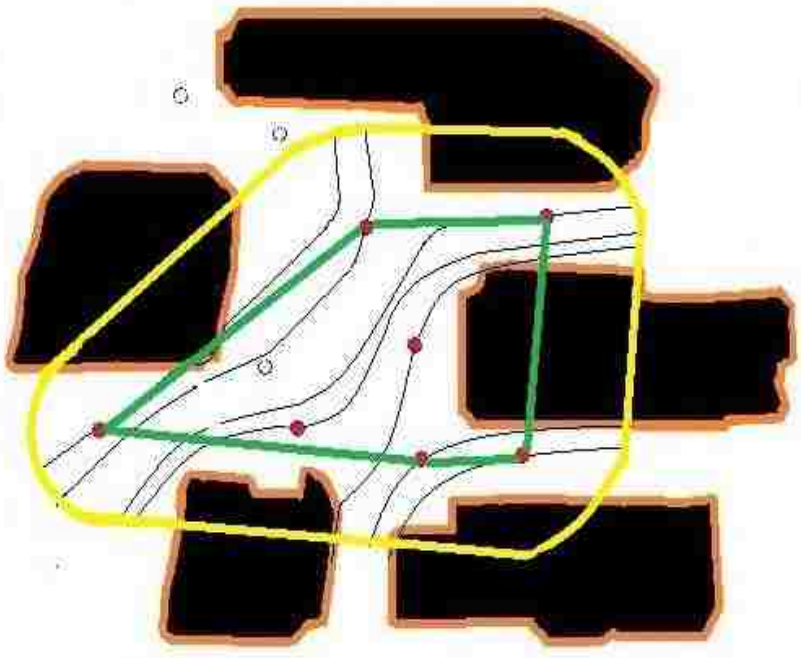


(e)

Figure 4.8. The credible sections of chemical path using different number of working sensor (cont.).

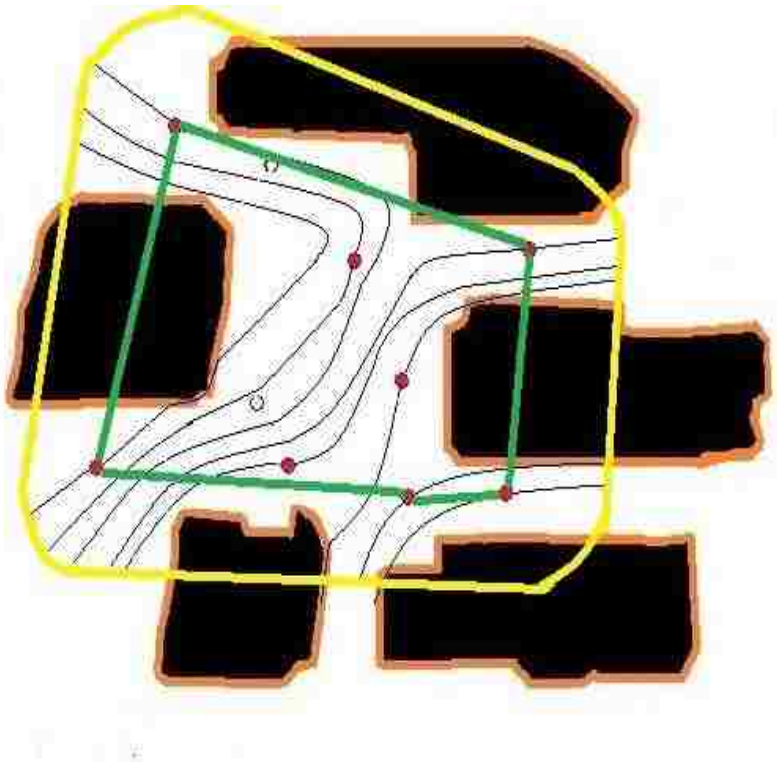


(f)



(g)

Figure 4.8. The credible sections of chemical path using different number of working sensor (cont.).



(h)

Figure 4.8. The credible sections of chemical path using different number of working sensor (cont.).

Especially, this software has CFD module. The CFD Module is the platform for simulating devices and systems that involve sophisticated fluid flow models. As is the case with all modules in the COMSOL Product Suite, the CFD Module provides ready-made physics interfaces that are configured to receive model inputs via the graphical user interface (GUI), and to use these inputs to formulate model equations. The particular physics interfaces that the CFD Module is equipped with enable you to model most aspects of fluid flow, including descriptions of compressible, noisothermal, non-Newtonian, two-phase, and porous media flows – all in the laminar and turbulent flow regimes. The CFD Module can be used as a standard tool for simulating computational fluid dynamics (CFD), or in collaboration with the other modules in the COMSOL Product Suite for Multiphysics simulations where fluid flow is important.

We assume that the wind is from southwest 45 degrees and the configuration is on same size and position with the Google map. This software is accorded to Finite Element method to process the fluid dynamic issue, so the author want to use the result from this software to do a comparison. From the results produced COMSOL, the author are able to obtain the velocity of airflow at every point and the particle paths. The author select ten points, and record its locations and velocities on these points. Figure 4.9 shows the steam line of airflow propagation which illustrates the chemical particles dispersion.

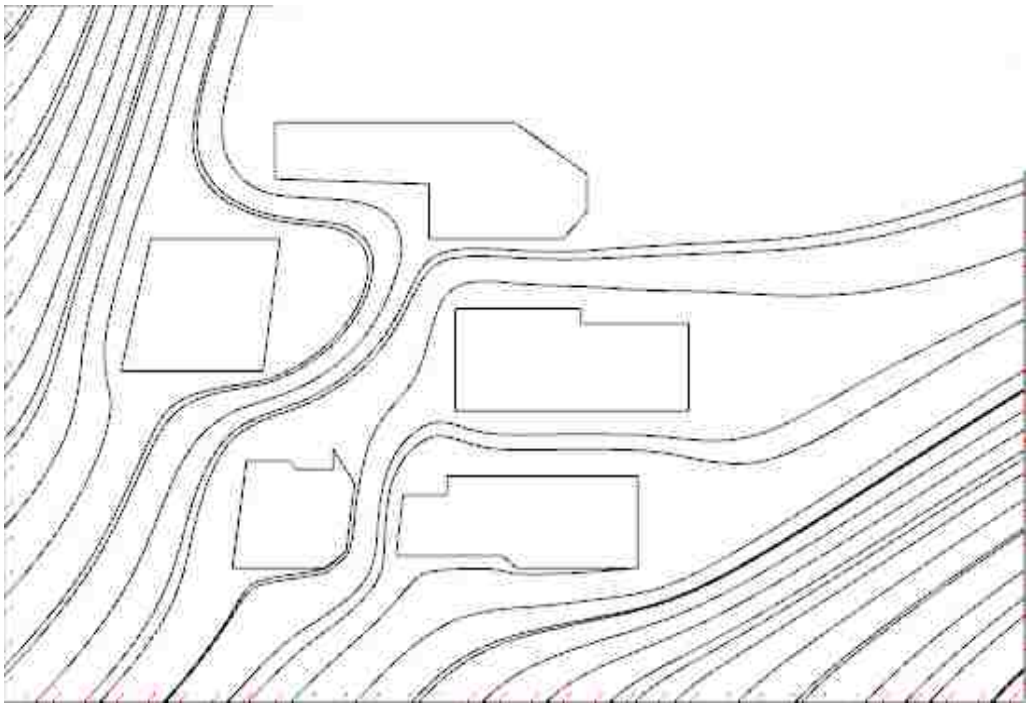


Figure 4.9. Air-borne particle paths going through ten sensors in a real map processed by COMSOL.

If the results using our proposed approach and the results using COMSOL are identical, it can support our proposed approach is correct. Our approach uses the same map, locations and velocities of sensors that are selected from the result of COMSOL. The circles signify the locations where the author fixed sensors. The red circles mean the sensors are working, and the blank circles mean the sensors are not working. Comparing the Figure 4.8 and Figure 4.9, the author can find the airflow path is almost same. Figure 4.8 shows the cases where different number of working sensors existed. When the working sensors are changed from eight to one, the maps of the airflows are different. A larger number of working sensor produce results closer to the real data obtained from COMSOL. The sensor's dispatch affects the result of simulation also. It is obvious that optimized dispatch can collect more information for the system. For an extreme example, if the author fix all of the sensors in a small area together, the effective paths will existed in this small area only.

4.3. CONCLUSION

There are many useful and humanitarian applications that can locate the source of a chemical source. Currently, the majority of work in this area uses reactive control schemes that track an odor plume along its entire length, which is slow and difficult in cluttered environments. This paper employs a high-level control scheme. The interpolation and extrapolation method is used to model the particle path in the sensors' environment. Then a reasoning system use the path model to get the velocity, chemical concentration at any point on the map and predict the most probable locations of the odor source. This approach has been shown to be effective for odor localization in a known environment, without the need for the robot to travel to the source.

With the further development there is great potential for this approach to lead to many valuable applications by generalization to a wider range of environmental configurations. The paper present development to solve the problem there exist obstacles and opening in the environment. The approach gives the mode of the particle path surrounding the obstacles and openings. The development has successfully applied in general environment, because the propagation of the chemical particle can go through obstacles and opening in actual case. In addition, this paper is the first example of using

interpolation and extrapolation method to model the particle path that applied in a real environment.

In addition, the result of simulating the particle path in the real map of campus is approximate with the result from the fluid dynamic analysis software, COMSOL. Hence, this approach can approximately map the particle path in real cases. Future work will concentrate on two areas of development: development of more general particle path modeling algorithm, and implementation of a range data acquisition system to enable autonomous map the building.

5. MULTI-SENSOR INTEGRATION TO MAP ODOR DISTRIBUTION FOR THE DETECTION OF CHEMICAL SOURCES

Last section addresses the problem of mapping the continuous particle paths which are most likely taken by the odor particle detected by the concentration sensors and anemometer sensors. The estimated particle paths are useful for odor source localization. However, when the sensor system collect s more information from the new imported sensors, the problem of updating particle paths and chemical particles propagation is faced to be solved. The simplest solution is using both primary sensors and new imported sensors to repeat the process that is mentioned at last section. In this way, the author need to redo the whole computation using a new sensor set. But, this way has a high computational cost and a more time consumption. Due to the new imported sensors are local, changing the whole particle paths and chemical particles propagation is not necessary. The author don't need to abandon the whole result from the previous computation. The only needed thing is updating the particle paths and chemical particles propagation around the new imported sensors.

In this section, a novel algorithm is proposed that smooth particle paths using the new imported sensors. In the real experiment, the new imported sensors are some micro-drones. The new imported sensors are scattered randomly in the considering area. They collect the new data about velocities, chemical concentrations, and the gradients of chemical concentrations at locations of new imported sensors. At the same times, depending on the particle paths and chemical particles concentration mapping that are derived by the primary sensors , the author can know the about velocities, chemical concentrations, and the gradients of chemical concentrations at locations of new imported sensors. Comparing the measured values from sensor and computational values from the above method, they are not identical. Hence, the particle paths and chemical particles concentration mapping need to be modified.

The new imported sensors are local, the data from these sensors only can estimate the particle paths and chemical particles concentration mapping around them. The author design a novel algorithm that modifies the parameters of particle paths around the new imported sensors. This novel algorithm uses the new imported sensors and the sensors that are adjoining the new sensors.

In the cases that more sensors are randomly scattered into the considered area for getting more precise particle paths, the author should propose the algorithm to use the additional sensors to smooth the particle paths.

This section addresses the problem of mapping odor distribution derived from a chemical source using multi-sensor integration and reasoning system design. Odor localization is the problem of finding the source of an odor or other volatile chemical. Most localization methods require a mobile vehicle to follow an odor plume along its entire path, which is time consuming and may be especially difficult in a cluttered environment. To solve both of the above challenges, this paper proposes a novel algorithm that combines data from odor and anemometer sensors, and combine sensors' data at different positions. Initially, a multi-sensor integration method, together with the path of airflow was used to map the pattern of odor particle movement. Then, more sensors are introduced at specific regions to determine the probable location of the odor source. Finally, the results of odor source location simulation and a real experiment are presented.

5.1. DETECTION OF ODOR SOURCE

The detection of airborne chemicals presents a different type of challenge than more traditional detection efforts, such as visual-based detection or propagating signal detection Chemicals that are airborne tend to drift in various directions due to wind, up-draft, and obstacles. As a result, isolation of the source of such particles becomes considerably difficult and dependent on topography and environment.

There has been some previous research on the detection and modeling of airborne particles, plume location and tracking. However, most of such research is based on sensor information on moving robots that are guided by the detectors. In [26], the author developed the model using naive physics airflow mapping. In [27], the odor localization used a bi-modal search with the complementary sensing of olfaction and vision. In [28], the author set up a mobile sensing system for localization of an odor source using gas and anemometric sensors. These types of sensing robots are assumed to move about freely following the trail of a chemical signature, while continuously searching for the particles. Both of these assumptions may be invalid in inaccessible and hostile environments with sensors that can either function one time or need long rejuvenation time cycles. To solve

these problems, the author proposed a novel algorithm of mapping continuous particle paths using discrete sensors for odor source localization, an application of a radial basis function neural network for chemical source detection, and odor source localization using spline interpolation with the complementary Hermite spline function neural network.

In our approach to the problem of chemical particle detection and source location, the author use a small number of chemical sensors that are sparsely scattered around an area only known by a two-dimensional map. In real-world problems, the author anticipate that an aircraft would drop some of these sensors on the area of interest while taking some aerial pictures. the author assume that the sensor data along with the map are transmitted to a nearby location, perhaps to a vehicle that will be traveling through the area of interest. the author would like to use the maximum available information content to generate first a model of the chemical particle distribution, and then locate the source of the particles based on the model. Because the author obtain the mapping of airflow by utilizing interpolation methods instead of finite element analysis, our approach saves time and computer processing. Finally, through a reasoning system, the author localize the area where the chemical source is located.

5.2. ODOR SENSOR AND ANEMOMETER SENSOR

The important aspects of detecting and tracking chemical sources are odor sensors and anemometer sensors. The odor sensors are for measuring the concentration of chemical particles, and the anemometer sensors are used for the direction of the airflow carrying chemical particles.

Over the last decade, "electronic sensing" or "e-sensing" technologies have undergone important developments from a technical and commercial point of view. The expression "electronic sensing" refers to the capability of reproducing human senses using sensor arrays and pattern recognition systems. Recent research has been conducted to develop technologies, commonly referred to as electronic noses that could detect and recognize odors and flavors [29]. The stages of the recognition process are similar to human olfaction and are performed for identification, comparison, quantification and other applications, including data storage and retrieval. These devices have undergone much development and are now used to fulfill industrial needs. The most commonly

available odor sensors detect the presence of airborne substances through changes in the electrical resistances of chemically sensitive carbon-doped polymer films.

An anemometer mounted on the sensor can provide relative velocity between the airflow and the anemometer. Wind speed and wind direction can be measured with a variety of tools. The most common, included with complete home weather stations, is the anemometer, which typically consists of a rotating vane to measure direction and a shaft with cups attached that spins with the wind to measure its speed. An anemometer looks like a weather vane, but instead of measuring which direction the wind is blowing with pointers, it has four cups so that it can more accurately measure wind speed. Each cup is attached to the end of a horizontal arm, each of which is mounted on a central axis, similar to the spokes on a wheel.

5.3. THE FRAMEWORK OF MULTI-SENSOR INTEGRATION

The framework of multi-sensors integration has two parts. First, it's the integration of the odor sensor and anemometer sensor. Second, it's the integration of the historical sensor and new sensor.

5.3.1. The Integration of the Odor Sensor & Anemometer Sensor. Particle-laden flow refers to a class of two phase fluid flow, in which one of phase is continuously connected (referred to as the continuous or carrier phase) and the other phase is made of small, immiscible and typically dilute particles (referred to as the dispersed or particle phase). The problem of detecting an odor source is typically about particle-laden flow. The chemical particle is the dispersed phase, and the air is the carrier phase.

An anemometer is a device used for measuring wind speed, and is a common weather station instrument. The term is derived from the Greek word anemos, which means wind, and is used to describe any wind speed measurement instrument used in meteorology. The first known description of an anemometer was given by Leon Battista Alberti in 1450.

A simple type of anemometer was invented in 1845 by Dr. John Thomas Romney Robinson, of Armagh Observatory. It consisted of four hemisphericalcups mounted on horizontal arms, which were mounted on a vertical shaft. The air flow past the cups in any horizontal direction turned the shaft at a rate that was proportional to the wind speed. Therefore, counting the turns of the shaft over a set time period produced a value

proportional to the average wind speed for a wide range of speeds. On an anemometer with four cups, it is easy to see that since the cups are arranged symmetrically on the end of the arms, the wind always has the hollow of one cup presented to it and is blowing on the back of the cup on the opposite end of the cross.

When Robinson first designed his anemometer, he asserted that the cups moved one-third of the speed of the wind, unaffected by the cup size or arm length. This was apparently confirmed by some early independent experiments, but it was incorrect. Instead, the ratio of the speed of the wind and that of the cups, the anemometer factor, depends on the dimensions of the cups and arms, and may have a value between two and a little over three. Every previous experiment involving an anemometer had to be repeated.

The three-cup anemometer developed by the Canadian John Patterson in 1926 and subsequent cup improvements by Brevoort & Joiner of the USA in 1935 led to a cup wheel design which was linear and had an error of less than 3% up to 60 mph (97 km/h). Patterson found that each cup produced maximum torque when it was at 45° to the wind flow. The three-cup anemometer also had a more constant torque and responded more quickly to gusts than the four-cup anemometer.

The three-cup anemometer was further modified by the Australian Dr Derek Weston in 1991 to measure both wind direction and wind speed. Weston added a tag to one cup, which causes the cup wheel speed to increase and decrease as the tag moves alternately with and against the wind. Wind direction is calculated from these cyclical changes in cup wheel speed, while wind speed is determined from the average cup wheel speed.

Three-cup anemometers are currently used as the industry standard for wind resource assessment studies & practice.

A chemical sensor is a device that transforms chemical information (composition, presence of a particular element or ion, concentration, chemical activity, partial pressure…) into analytically useful signal. The chemical information, mentioned above, may originate from a chemical reaction of the analytic or from a physical property of the system investigated. They can have applications in different areas such as medicine, home safety, environmental pollution and many others.

Chemical sensors usually contain two basic components connected in series: a chemical (molecular) recognition system (receptor) and a physicochemical transducer. In the majority of chemical sensors, the receptor interacts with analytic molecules. As a result, its physical properties are changed in such a way that the appending transducer can gain an electrical signal.

Receptor: The function of the receptor is fulfilled in many cases by a thin layer which is able to interact with the analytic molecules, catalyze a reaction selectively, or participate in a chemical equilibrium together with the analytic. The receptor layer can respond selectively to particular substances or to a group of substances. The term molecular recognition is used to describe this behavior. Among the interaction processes, the most important for chemical sensors are adsorption, ion exchange and liquid-liquid extraction. Primarily these phenomena act at the interface between analytic and receptor surface.

Transducer: Nowadays, signals are processed almost exclusively by means of electrical instrumentation. Accordingly, every sensor should include a transducing function, i.e. the actual concentration value, a non-electric quantity must be transformed into an electric quantity, voltage, current or resistance. Some of them develop their sensor function only in combination with an additional receptor layer. In other types, receptor operation is an inherent function of the transducer.

If the mass fraction of the dispersed phase is small, one-way coupling between the two phases is a reasonable assumption; that is, the dynamics of particle phases are affected by the carrier phase, but the reverse is not the case. In our case, the particles are very small and occur in low concentrations; hence the dynamics are governed by the carrier phase. The particle phase is typically treated in a Gaussian distribution [30] along the flow direction

$$C(x,y) = \frac{q}{2\pi k d_s} \exp[-\frac{u}{2K}(d_s - \Delta d)^2],$$

where $\mathrm{d}_s = \sqrt{(x_s - x)^2 + (\mathrm{y}_s - y)^2}$, $\Delta d = (x_s - x)\cos\theta + (y_s - y)\sin\theta$.

C is the concentration (ppm), q is the emitted rate (mL/s), u is the wind speed (m/s), K is turbulent diffusion coefficient (m2/s), and θ is the angle from the x-axis to the upwind direction.

Figure 5.1 shows the two paths generated by matching the expected and sensed concentration values, as well as the initial and final velocities.

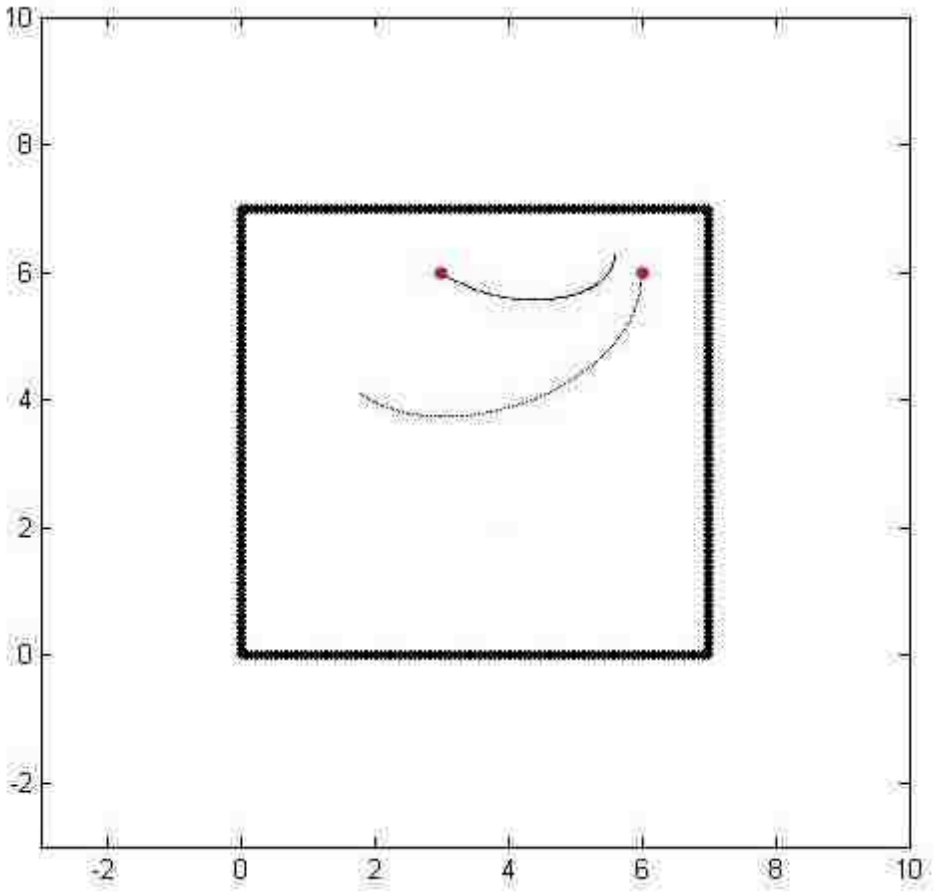


Figure 5.1. The two paths generated by matching the expected and sensed concentration values.

The second issue is related to the choice of the parameter t. In our parametrization, the author chose t to start at 0 at one of the sensors and end at 1 at another sensor. The author would like to have the parameter be a good representation of actual travel time, since the author also would like to obtain connected paths. To correct this problem, the author compute the speed at every point along the path as a linear function of the distance from one sensor to the other one while matching the sensed speed values at the two end points. Figure 5.2 shows the two paths with equally timed distances.

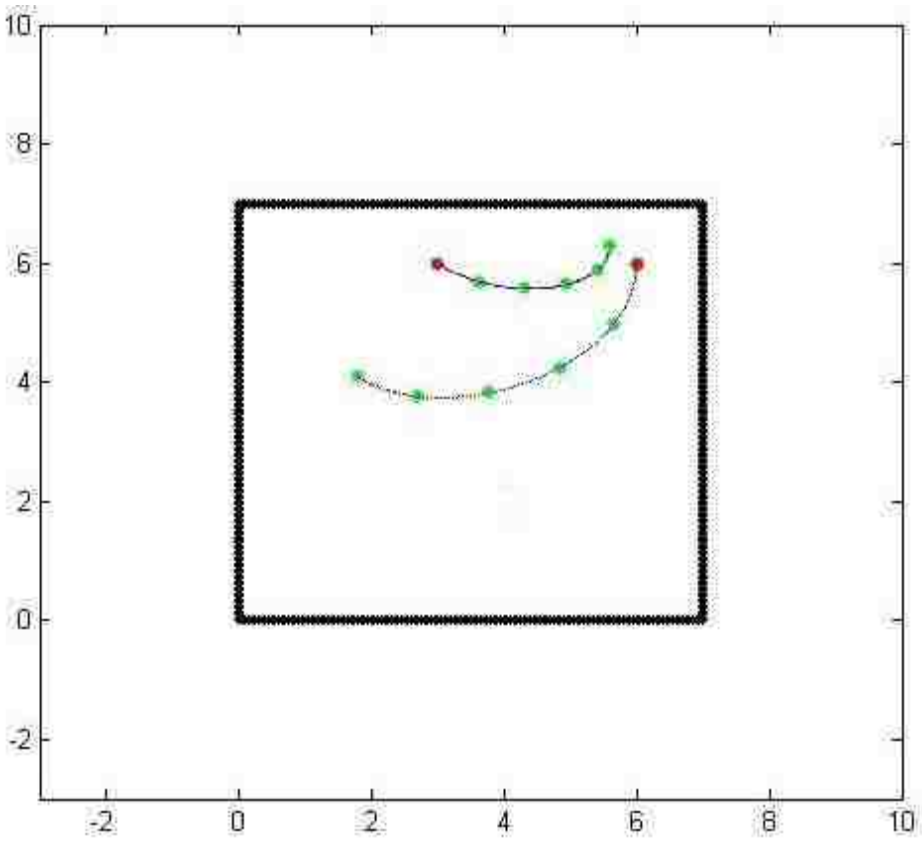


Figure 5.2. The two paths generated by matching the expected and sensed concentration values with equally timed distances.

Even though the author now have a path from one sensor to another, there are still several issues to be resolved. The first issue is related to the underlying presumption that a particle would somehow travel from one sensor to the other even though the sensors are at arbitrary locations. To correct this problem, the author rely on the dissipation property of the particles. The author compute the expected concentration value along the computed path and compare it with the actual sensed concentration value. Based on the error and the measured gradient concentration, the author determine a new location perpendicular to the initial path where the expected and sensed concentration values match. The author then compute the corrected path going through one of the sensors and the new location. When the author repeat the process forwards from one sensor and backwards from another one, the author end up getting two consistent paths with correct concentration values.

5.3.2. The Integration of the History Sensors & New Sensors. When more sensors are introduced to the region, the author need to incorporate the new data and update the particle flow paths. The author can integrate the data from the newly added sensors by processing the complete set of sensor data, or the author can update the existing air flow paths in the neighborhoods of the new sensors. In this paper, the author utilize a novel approach to update the particle paths described by the interpolation functions.

In the original particle path calculations, the author generated some primary paths that go through each of the original sensor location and match the sensed values of the particle concentrations. When new sensors are added in between theses primary paths, the author need to interpolate and determine secondary paths that go through the new sensors. Since the particle concentration values on these interpolated secondary paths don’t necessary match the observed values from the new sensors, the author need to update the primary path data as well.

As a test case, the author initially place 3 sensors and obtain the primary paths from the sensed values. Using the primary paths, the author can get secondary paths that map the whole considered area, as shown in Figure 5.3. The author then place another sensor inside the region of interest. Naturally, the sensed values at the new sensor

doesn't match the extrapolated values based on the perpendicular extensions from the primary paths exactly.

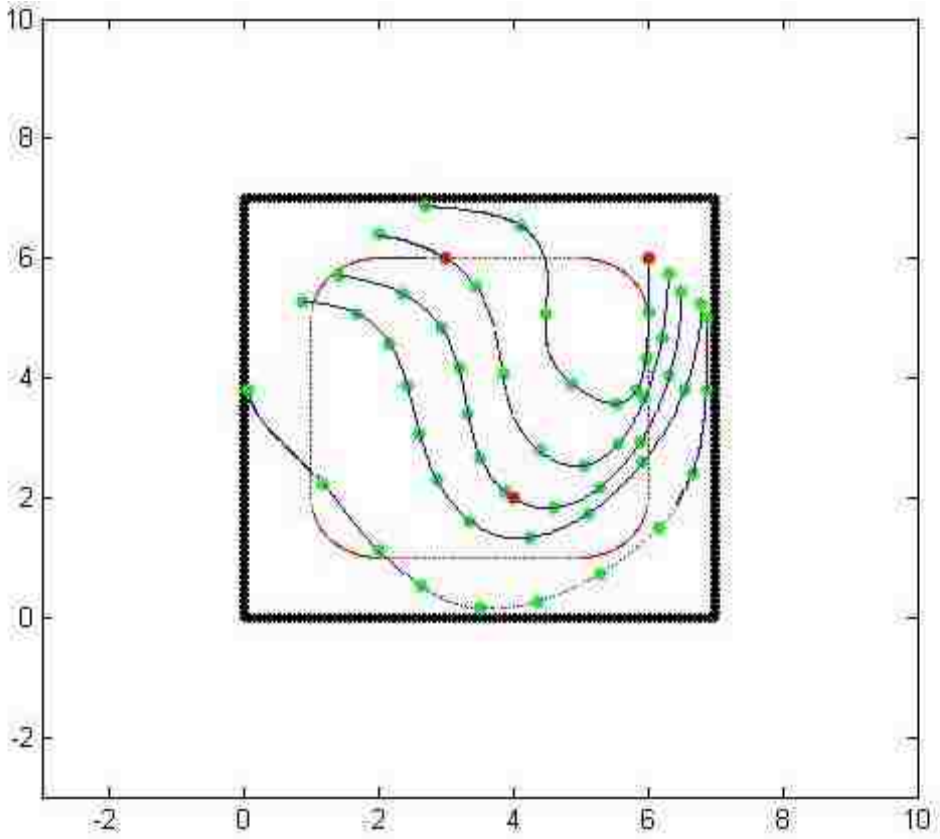


Figure 5.3. Primary air-borne particle paths going through three sensors and an additional sensor.

When the new sensor is placed at (x_*, y_*), the author only need to update the relevant primary paths. The location of the new sensors in relation to the neighboring primary paths determines the paths to be changed. The region where is needed to be changed is defined by the closest primary paths and the perpendicular lines through the sensors on the primary paths. The author denote the two sensors on these primary paths

as (x_i, y_i) and (x_j, y_j). The author model the odor propagation between two primary sensors as two connected particle paths that relate the two primary sensors and the additional sensor. The author then use our method to update the parameters of the primary paths to join the piecewise particle paths.

The updated path equations are modeled by two pieces of particle paths.

The first path, denoted by L_1, is from (x_i, y_i) to (x_*, y_*), such that

$$\begin{aligned} x(t) &= (2(x_i - x_*) + 3(\delta x_i + \delta x_*))t^3 + 3(x_* - x_i) - 3(\delta x_* + 2\delta x_i))t^2 + 3\delta x_i t + x_i \\ y(t) &= (2(y_i - y_*) + 3(\delta y_i + \delta y_*))t^3 + 3(y_* - y_i) - 3(\delta y_* + 2\delta y_i))t^2 + 3\delta y_i t + y_i \end{aligned} \quad (1)$$

The piecewise connected second path, denoted by L_2 , is from (x_*, y_*) to (x_j, y_j), such that

$$\begin{aligned} x(t) &= (2(x_* - x_j) + 3(\delta x_* + \delta x_j))t^3 + 3(x_j - x_*) - 3(\delta x_j + 2\delta x_*))t^2 + 3\delta x_* t + x_* \\ y(t) &= (2(y_* - y_j) + 3(\delta y_* + \delta y_j))t^3 + 3(y_j - y_*) - 3(\delta y_j + 2\delta y_*))t^2 + 3\delta y_* t + y_* \end{aligned} \quad (2)$$

Comparing the particle paths between the updated path and two pieces of particle paths, the author define the error term as

$$E(x, y) = \frac{1}{2}\sqrt{(\int_{L_1} y \cdot dx/dt\, dt + \int_{L_2} y \cdot dx/dt\, dt - \int_{\hat{L}} y \cdot dx/dt\, dt)^2} \quad (3)$$

where $\hat{L}$ indicates it's the path without the additional sensor.

In Equation (3),

$$\begin{aligned} \int_{\hat{L}} y \cdot dx/dt \,\mathrm{dt} = \int_i^j & ((2(y_i - \hat{y}_j) + 3(\delta y_i + \delta \hat{y}_j))t^3 + 3(\hat{y}_j - y_i) - 3(\delta \hat{y}_j + 2\delta y_i))t^2 \\ & + 3\delta y_i t + y_i) \cdot ((2(x_i - \hat{x}_j) + 3(\delta x_i + \delta \hat{x}_j))t^3 + 3(\hat{x}_j - x_i) \\ & - 3(\delta \hat{x}_j + 2\delta x_i))t^2 + 3\delta x_i t + x_i)' dt. \end{aligned} \quad (4)$$

where $(\hat{x}_j, \hat{y}_j \cos^{-1}\theta)$ represents the updated location for the jth sensor (x_j, y_j), and the path $\hat{L}$ is from (x_i, y_i) to $(\hat{x}_j, \hat{y}_j)$.

When the author substitute Equations (1), (2) and (4) into Equation (3), the author get an equation in terms of the unknown parameters $\hat{x}_j$, $\hat{y}_j$, $\delta\hat{x}_j$, and $\delta\hat{y}_j$. The author assume that the velocity variables $\delta\hat{x}_j$ and $\delta\hat{y}_j$ are preserved, and the updated endpoint $(\hat{x}_j, \hat{y}_j)$ is on the perpendicular line to the primary path going through the jth sensor.

Based on Equation (3), the minimization of the error can be determined by $\arg\min_{\hat{x}_j,\hat{y}_j} E(\hat{x}_j,\hat{y}_j)$, the arguments of the minimum is the point at which the error term E attains its smallest value. The most ideal situation is when $\int_{L_1} y \cdot dx/dt \, \mathrm{dt} + \int_{L_2} y \cdot dx/dt \, \mathrm{dt} = \int_{L} y \cdot dx/dt \, \mathrm{dt}$, where the error becomes zero. The expression of $\arg\min_{\hat{x}_j,\hat{y}_j} E(\hat{x}_j,\hat{y}_j)$ is

$$\arg\min_{\hat{x}_j,\hat{y}_j} E(\hat{x}_j,\hat{y}_j) = \left\{ (\bar{x}_j,\bar{y}_j) \middle| E(\bar{x}_j,\bar{y}_j) \le E(\hat{x}_j,\hat{y}_j) \right\}. \tag{5}$$

We use $(\bar{x}_j,\bar{y}_j)$ to denote the optimal parameters. Since $(\bar{x}_j,\bar{y}_j)$ is on the perpendicular line to the primary path going through the sensor, $(\bar{x}_j,\bar{y}_j)$ satisfies

$$\bar{y}_j = \frac{\delta y_j - y_j}{\delta x_j - x_j}(\bar{x}_j - x_j) + y_j. \tag{6}$$

Therefore, the updated law is such that

$$\begin{aligned} \bar{x}_j &= \arg\min_{x_j,y_j} E(x_j,y_j) \\ \bar{y}_j &= \frac{\delta y_j - y_j}{\delta x_j - x_j}(\bar{x}_j - x_j) + y_j \\ \delta\bar{x}_j &= \delta x_j \\ \delta\bar{y}_j &= \delta y_j \end{aligned} \tag{7}$$

Using $(\bar{x}_j,\bar{y}_j)$ to substitute $(\hat{x}_j,\hat{y}_j)$, the author can get the updated forward particle path as

$$\begin{aligned} x(t) &= (2(x_i - \bar{x}_j) + 3(\delta x_i + \delta\bar{x}_j))t^3 + (3(\bar{x}_j - x_i) - 3(\delta\bar{x}_j + 2\delta x_i))t^2 + 3\delta x_i t + x_i \\ y(t) &= (2(y_i - \bar{y}_j) + 3(\delta y_i + \delta\bar{y}_j))t^3 + (3(\bar{y}_j - y_i) - 3(\delta\bar{y}_j + 2\delta y_i))t^2 + 3\delta y_i t + y_i. \end{aligned} \tag{8}$$

Similarly, the author can get the backward particle paths using the same method to update the variables for the ith sensor. Figure 5.4 shows the updated map of the particle paths.

One of the important applications of chemical sensing is odor localization. This is the act of finding the location of a chemical source in the environment and is the topic of this paper. Odor localization is performed by many type of mechanism, in a variety of

methods. Sensing and responding to chemical is now a burgeoning area of robotics research.

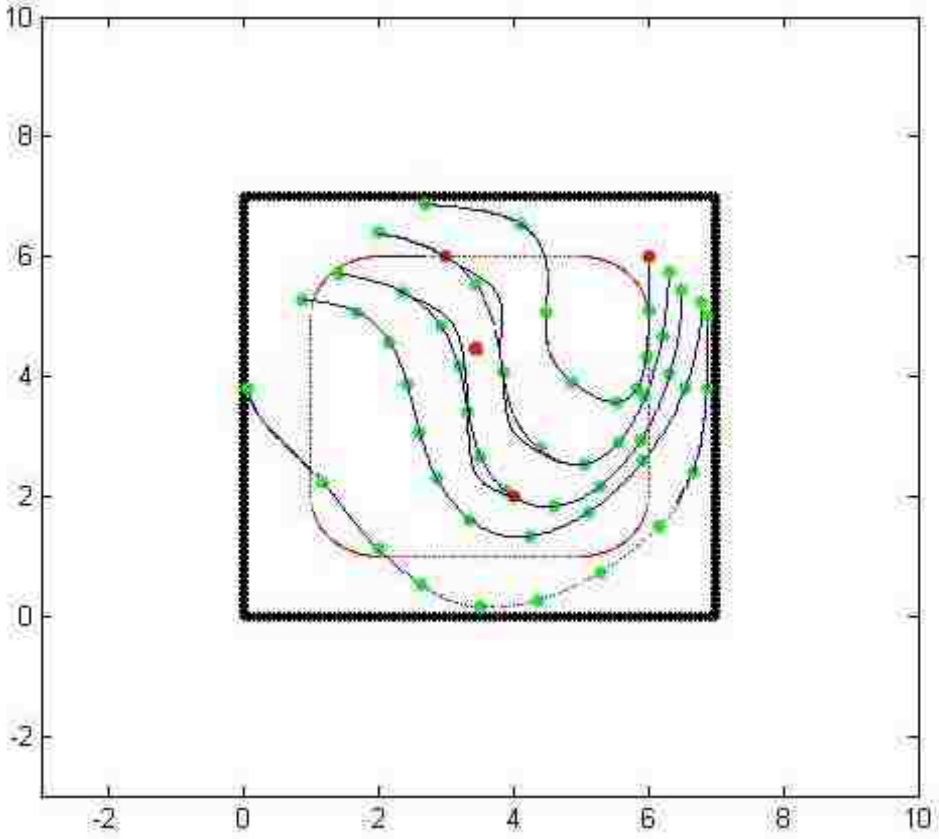


Figure 5.4. Updated air-borne particle paths going through three sensors and an additional sensor.

The ability to locate of an odor/chemical source has many valuable applications. These applications include finding the source of dangerous substances such as airborne biological material, hazardous chemical, gas and other pollutants, in industrial and other settings; searching for survivors in earthquake-damaged buildings; detecting fire in its initial stages; locating unexploded mines and bombs. These applications have been

tackled by researchers in a variety of methods. Most of their works have focused on mobile robots to detect plume and trace plume. Odor dispersal occurs through carriage by the fluid current. Many researchers has used a chemical gradient method that is acquiring the plume firstly, and then moving upwind along the plume. But, their limitations are that robots must follow the plume along its entire length, which is time consuming and may not be possible in some environment where a well-defined plume does not be formed. In addition, the effects of objects and the walls on both airflow and robot mobile are often neglected. In this paper, the author have attempted to overcome these limitations. By exploiting more information from the environment in addition to local wind and chemical reading, the airflow map and chemical distribution can be derived first, then basing the aforementioned result a „sense-map-plan-act" control scheme is built. However, when the sensor system collects more data, the problem of updated mapping is faced to be solved. Because the information is local, which does not affect the whole mapping, the author just need update the airflow and chemical dispersion around the sensors that gather new data. Neural networks using splines function can update its weights to satisfy the new data from sensors.

In this section, the theory of spline approximation is adequately combined with the neural network principle by these advantages of spline function, as shown in Figure 5.5.

The neural network model based on m times spline basis function is constructed, where w_j is the weight of neural network and the $\phi_j(t)$ is spline basis function. That is $\phi_j(t) = t^j, j = 0,1,\cdots,m,$ is the hidden layer neuron incentive function $\mathrm{t} \in [0,1]$. Suppose the weight matrix $W = [w_o, w_1, \cdots, w_m]^T$ and the incentive matrix is $\Phi(t) = [\phi_o, \phi_1, \cdots, \phi_m]^T$. So the output of the neural network is $X(\mathrm{x}, \mathrm{y}) = \sum_{j=0}^{m} w_j \phi_j(t) = W^T \Phi(\mathrm{t})$. Comparing with the cubic Hermite splines, the weights of the neural networks are the parameters of the splines function.

The error function is $e(\mathrm{k}) = X^{'}(t_n) - f^{'}(t_n), n = 0,1,\cdots N+2,$ where $n = N+2$, N is the number of new sensor points, $f^{'}(t_n)$ is from the new data and $X^{'}(t_n)$ is from the

spline function. Therefore, the error function is the difference between splines interpolation and the real data.

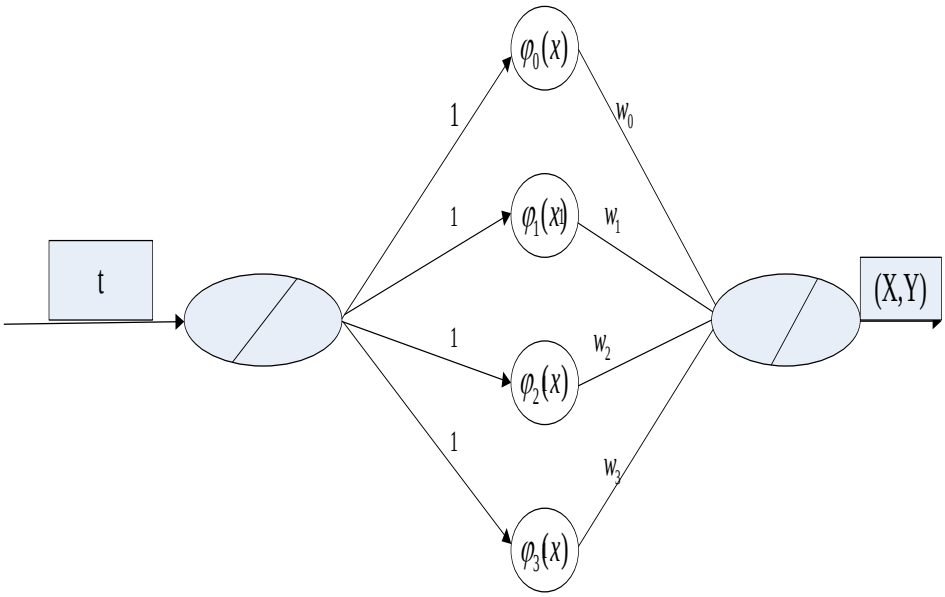


Figure 5.5. Neural network model based on spline basic function.

The performance index is $J = \frac{1}{2}\sum_{k=0}^{n} e^2(k)$. and the weight adjustment is $W(\mathrm{k}+1) = W(\mathrm{k}) + \eta J(\mathrm{k})\Phi(\mathrm{k})$.

5.4. CONVERGENCE THEOREM OF NEURAL NETWORK MODEL

Theorem 3: Let η be the learning rate. Then the neural network algorithm is convergent, when $0 \le \eta \le \frac{2}{\sum_{j=0}^{m} \left|\phi_j(t)\right|^2}$, where m is the number of hidden layer neurons.

Proof: Let the Lyapunov function $V(\mathrm{k}) = \frac{1}{2} e^2(k)$.

Then $\Delta V(k) = \frac{1}{2} e^2(k+1) - \frac{1}{2} e^2(k)$,

And $e(k+1) = e(k) + \Delta e(k) = e(k) + (\frac{\partial e(k)}{\partial W})^T \frac{\partial e(k)}{\partial W}$,

$\Delta W = -\eta e(k) \frac{\partial e(k)}{\partial W}$,

So,

$\Delta e(k) = -\eta e(k) (\frac{\partial e(k)}{\partial W})^T \frac{\partial e(k)}{\partial W} = -\eta e(k) \left\| \frac{\partial e(k)}{\partial W} \right\|_2^2$,

Where $\|\cdot\|_2 = \sqrt{\sum |.|^2}$ is the Euclidean norm. So Lyapunov function can be written as

$$\Delta V(k) = \frac{1}{2}[\mathrm{e(k)} + \Delta \mathrm{e(k)}]^2 - \frac{1}{2} e^2(k)$$

$$= \Delta \mathrm{e(k)}[\mathrm{e(k)} + \frac{1}{2} \Delta \mathrm{e(k)}]$$

$$= -\eta e(k) \left\| \frac{\partial e(k)}{\partial W} \right\|_2^2 [\mathrm{e(k)} - \frac{1}{2} \eta e(k) \left\| \frac{\partial e(k)}{\partial W} \right\|_2^2]$$

$$= \left\| \frac{\partial e(k)}{\partial W} \right\|_2^2 \mathrm{e(k)}(-\eta + \frac{1}{2} \eta^2 \left\| \frac{\partial e(k)}{\partial W} \right\|_2^2)$$

To make the neural network algorithm convergent, the author have

$-\eta + \frac{1}{2} \eta^2 \left\| \frac{\partial e(k)}{\partial W} \right\|_2^2 \le 0, \eta > 0$,

That is $0 < \eta \le \frac{2}{\left\| \frac{\partial e(k)}{\partial W} \right\|_2^2}$.

We get,

$\frac{\partial e(k)}{\partial W} = (\frac{\partial e(k)}{\partial X(t)})(\frac{\partial X(t)}{\partial W}) = -\Phi(\mathrm{t})$.

According to $\Phi(t) = [\phi_o, \phi_1, \cdots, \phi_m]^T$, it can be proved

$$\left\| \frac{\partial e(k)}{\partial W} \right\|_2^2 = \left\| -\Phi(\mathrm{t}) \right\|_2^2 = \sum_{j=0}^{m} \left| \phi_j \right|^2 .$$

when $0 \le \eta \le \dfrac{2}{\sum_{j=0}^{m} \left| \phi_j(t) \right|^2}$, the author have $\Delta V < 0$.

Consequently, it is shown that the neural network algorithm is convergent.

5.5. ALGORITHM OF NEURAL NETWORK MODEL

As a test case, the author assume to use n sensors to collect information in the environment. In Figure 5.6, however, when the author make one more sensor in the search are, the aforementioned method of mapping can only be changed around the new sensor.

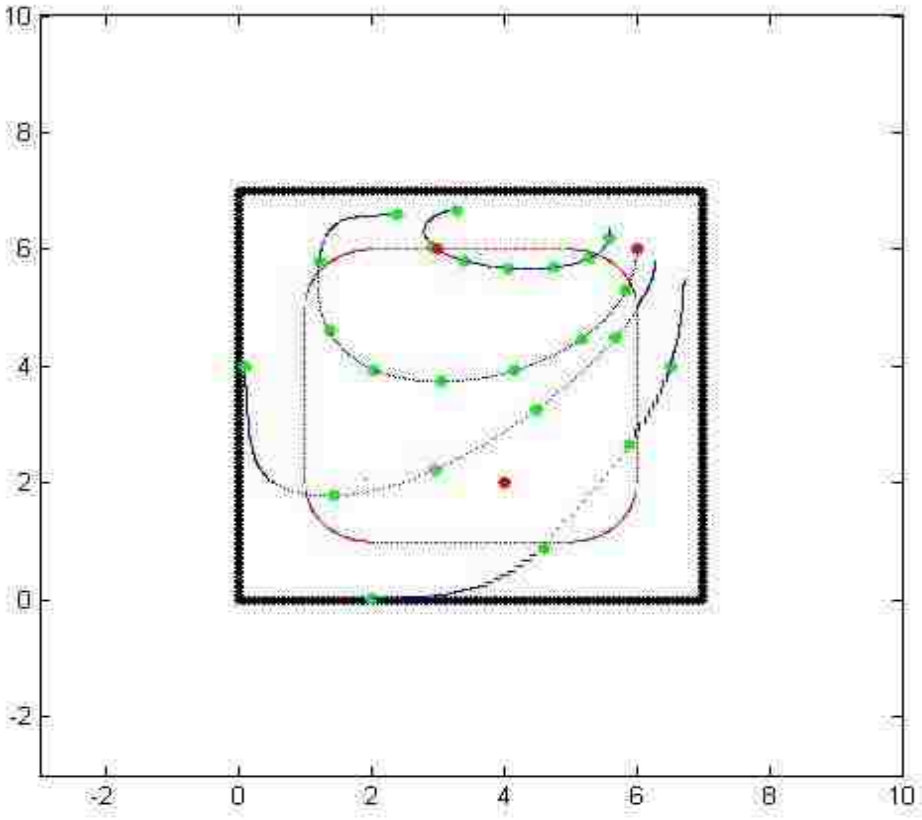


Figure 5.6. The particle paths going through two primary sensors.

According to the neural network model based on the slpine function which is discussed above, the author get the following neural network algorithm for update the airflow path:

Take the learning rate satisfying theorem, to make sure convergence of the neural network;

Calculate the output $f^{'}(t_n)$ t of neural network:

$$X(x,y)=\sum_{j=0}^{m} w_j\phi_j(t)=W^T\Phi(t);$$

Calculate the error function:

$$e(\mathrm{k})=X^{'}(t_n)-f^{'}(t_n), n=0,1,\cdots N+2,$$

Where, $X^{'}(t_{n,k})$ demonstrate the gradient of the particle path, and $f^{'}(t_n)$ is given by the real data;

Calculate the performance index of the neural network:

$$J=\frac{1}{2}\sum_{k=0}^{n} e^2(k);$$

Adjust the neural network weight: $W(\mathrm{k}+1)=W(\mathrm{k})+\eta J(\mathrm{k})\Phi(\mathrm{k})$.

In the Figure 5.6, the top two sensors are primary sensor. Using the data from these sensors the author can get the airflow path and chemical dispersion that are shown in the figure. The bottom sensor is not used in this case.

When the new data from the bottom sensor, the author can consider neural network method that can get the updated map by training the weights of the network. In this case, the t is considered to the input and the position (x,y) is considered to the output, and the error term is the difference between spline functions that are derived from output and the real data from sensors. In the Figure 5.7, the new map using the new data from the bottom sensor is shown below.

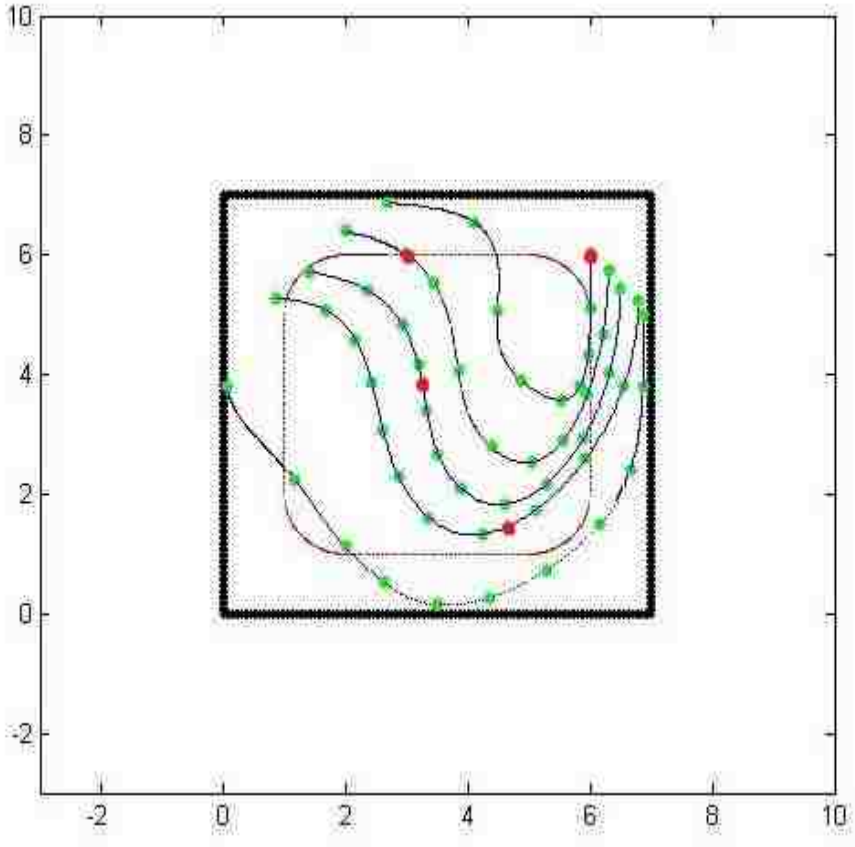


Figure 5.7. The new particle path map using 3 sensors.

5.6. CONCLUSION

There are many useful and humanitarian applications that can locate the source of a chemical source. Currently, the majority of work in this area uses reactive control schemes that track an odor plume along its entire length, which is slow and difficult in cluttered environments. This paper employs a high-level control scheme. The interpolation and extrapolation method is used to model the particle path in the sensors' environment. Then a reasoning system use the path model to get the velocity, chemical concentration at any point on the map and predict the most probable locations of the odor

source. This approach has been shown to be effective for odor localization in a known environment, without the need for the robot to travel to the source.

With the further development there is great potential for this approach to lead to many valuable applications by generalization to a wider range of environmental configurations. The paper present development to solve the problem there exist obstacles and opening in the environment. The approach gives the mode of the particle path surrounding the obstacles and openings. The development has successfully applied in general environment, because the propagation of the chemical particle can go through obstacles and opening in actual case. In addition, this paper is the first example of using interpolation and extrapolation method to model the particle path that applied in a real environment.

6. DETECTION AND TRACKING OF AN ODOR SOURCE IN SENSOR NETWORKS USING A REASONING SYSTEM

6.1. INTRODUCTION

The detection of the airborne chemicals presents a different type of challenge than the more traditional detection efforts, such as the visual-based detection or propagating signal detections. The chemicals that are airborne tend to drift in various directions due to wind, up-draft, and obstacles. As a result, isolation of the source of such particles becomes considerable difficult and dependent on topography and environment. There has been some previous research on the detection and modeling of airborne particles, plume location and tracking. However, most of such research is based on sensor information on moving robots that are guided by the detectors. These types of sensing robots are assumed to move about freely following the trail of a chemical signature, while they're continuously sensing for the particles. Both of these assumptions are invalid in inaccessible and hostile environments with sensors that can either function once or need along rejuvenation time cycles. In our approach to the problem of chemical particle detection and source location, the author use a small number of chemical sensors that are sparsely scattered around an area only known by a two-dimensional map. In real-world problems, the author anticipate that a small unmanned aircraft would drop some of these sensors on the area of interest while taking some aerial pictures. the author assume that the sensor data along with the map are transmitted to a nearby location perhaps to a vehicle that will be travelling through the area of interest. The author would like to use the maximum available information content to generate first a model of the chemical particle distribution, and then locate the source of the particles based on the model.

6.2. PARTICLE PATHS MAPPING AND ODOR DISPERSAL

A. Particle path algorithms using interpolation and extrapolation.

Using the sensors that can collect the sensors position, wind velocity, chemical concentration, and the author can determine the particle paths that describe the propagation in the environment. This map is a prerequisite for the detection the odor source.

In this paper, the author start with the interpolation of two nodes points (x_0, y_0) and (x_1, y_1), where the points are the locations of two sensors with odor particle values of s_0 and s_1, respectively. Since a direct interpolation of a path between the two points would be inconsistent with the odor propagation and the air flow, the author generate two more localizations, a propagation parameter "t" where $0 \le t \le 1$, and consistent interpolation functions H_x and H_y, such that

$$(x(t), y(t)) \approx (H_x(t), H_y(t)),$$

where $x_0 = H_x(0),\ \ x_1 = H_x(1),\ \ y_0 = H_y(0),\ \ y_1 = H_y(1).$

In this approximation, the author use Hermite polynomials. In the above equation, the author match the boundary values of the location; however the author also need to match the velocities $\partial x_0 / \partial t,\ \partial x_1 / \partial t,\ \partial y_0 / \partial t,$ *and* $\partial y_1 / \partial t.$

From the sensor data, the author can only collect the derivatives of y with respect to t, but the author need the derivatives of x and y with respect to t. However, these derivatives aren't too hard to determine from using the identity

$$\partial y / \partial x = \frac{\partial y / \partial t}{\partial x / \partial t}$$

Consequentially, the author chose

$$\frac{\partial x}{\partial t}\Big|_{t=0} = \frac{\partial x_0}{\partial t} = \delta x_0,$$
$$\frac{\partial y}{\partial t}\Big|_{t=0} = \frac{\partial y_0}{\partial t} = \delta y_0,$$
$$\frac{\partial x}{\partial t}\Big|_{t=1} = \frac{\partial x_1}{\partial t} = \delta x_1,$$
$$\frac{\partial y}{\partial t}\Big|_{t=1} = \frac{\partial y_1}{\partial t} = \delta y_1.$$

We, then, proceed to construct the two Hermite polynomials in the usual way, such that

$$
\begin{aligned}
H_x(t) &= ([1-2(t-0)L'_{1,0}(0)]L_{1,0}(t)^2)x_0 \\
&\quad +((t-0)L_{1,0}(t)^2)(\delta x_0) \\
&\quad +([1-2(t-1)L'_{1,1}(1)]L_{1,1}(t)^2)x_1 \\
&\quad +((t-1)L_{1,1}(t)^2)(\delta x_1) \\
&= ([1-2(t-0)(-1)(\frac{t-1}{0-1})^2)x_0 \\
&\quad +((t-0)(\frac{t-1}{0-1})^2)(x_0^+ - x_0) \\
&\quad +([1-2(t-1)(1)](\frac{t-0}{1-0})^2)x_1 \\
&\quad +((t-1)(\frac{t-0}{1-0})^2)(\delta x_1) \\
&= (1+2t)(t-1)^2 x_0 + t(t-1)^2(\delta x_0) \\
&\quad +(3-2t)t^2 x_1 + (t-1)t^2(\delta x_1)
\end{aligned}
$$

,where $L_{n,j}$ denotes the jth Lagrange coefficient of the $2n+1$ is the order polynomial.

Similarly, the author have

$$
\begin{aligned}
H_y(t) &= (1+2t)(t-1)^2 y_0 + t(t-1)^2(\delta y_0) \\
&\quad +(3-2t)t^2 y_1 + (t-1)t^2(\delta y_1)
\end{aligned}
$$

As a test case, the author consider a three sensor configuration system as in Figure 6.1. In the figure, the thick black lines are the boundaries of the room, the red dots are the sensor locations, and the red dotted lines designate the border of the boundary zone.

Some chemical sensors are designed to detect simply the existence of chemical particles and trigger a positive result when the concentration amounts are above a preset threshold level. In our design, instead of the threshold, the author make use of the actual concentration levels that are detected. This approach along with some other data enables us to model the flow of the particles and the location of the source. Each sensor provides the co-located sensory information of the airflow information that is obtained not by an additional sensory device but by an off-centered multi-orifice detection hardware configuration. In our derivations, the author assume that the differential information is perpendicular to the wind direction, but the author can accommodate any non-zero

known angular orientation simply by a coordinate transformation. Designating the location of the sensors by (x, y), the author represent the flow of air by (δx, δy). Similarly, the author represent the sensed particle concentration by s and the concentration gradient by δs.

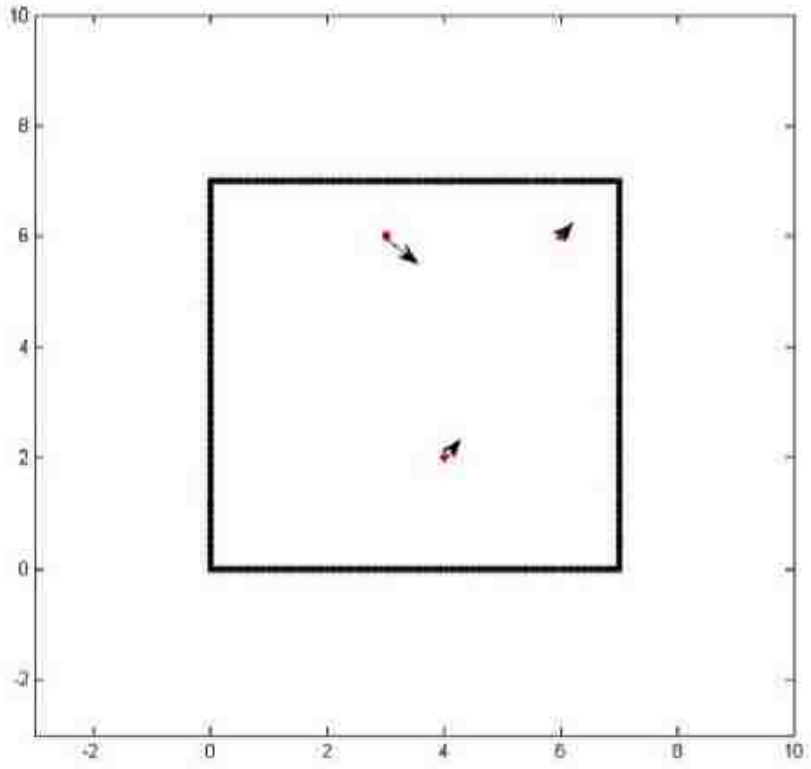


Figure 6.1. Three sensor configuration system.

Once the author obtain the sensory information, the author start with an approximation of the particle path. The author configure paths that go through the sensor locations, such that the paths satisfy the locations as well as the differentials. This

approach leads to a parametric cubic-polynomial representation of the path in terms of the variable t. the author use the cubic Hermite splines with the end point differentials weighted three times, such that

$$x(t) = (2(x(0) - x(1)) + (\delta x(0) + \delta x(1)))t^3 + 3(x(1) - x(0)) - (\delta x(1) + 2\delta x(0)))t^2 + \delta x(0)t + x(0),$$

$$y(t) = (2(y(0) - y(1)) + (\delta y(0) + \delta y(1)))t^3 + 3(y(1) - y(0)) - (\delta y(1) + 2\delta y(0)))t^2 + \delta y(0)t + y(0),$$

where the parametric curve starts at one sensor location at (x(0), y(0)) and ends at the other sensor location at (x(1), y(1)) as t goes from 0 to 1. Figure 6.2 shows the two paths generated by matching the expected and the sensed concentration values.

We compute the expected concentration values along the computed path and compare them with the actual sensed concentration values Based on the errors and the measured gradient concentrations; the author determine new locations perpendicular to the initial paths, where the expected and the sensed concentration values match. We, then, compute the corrected paths going through one of the sensors and the new location. When the author repeat this process forwards from one sensor and backwards from another, the author end up getting two consistent paths with correct concentration values. The author will refer to these paths as primary paths.

In the next step of the extrapolation, the author complete the particle propagation paths by generating secondary paths for the whole area. The secondary paths are between two adjacent primary paths. To generate these secondary paths, the author determine the perpendicular lines to the tangents of the paths, and use the intersection points of these perpendicular lines. The author assign the average values of the particle concentrations and the concentration gradients on the secondary paths. For the paths that are on the external regions of the primary paths, the author use perpendicular normal extensions, but the author extrapolate the particle concentrations and the concentration gradients. Figure 6.3 shows the path extensions as well as the whole room coverage with primary and the secondary paths.

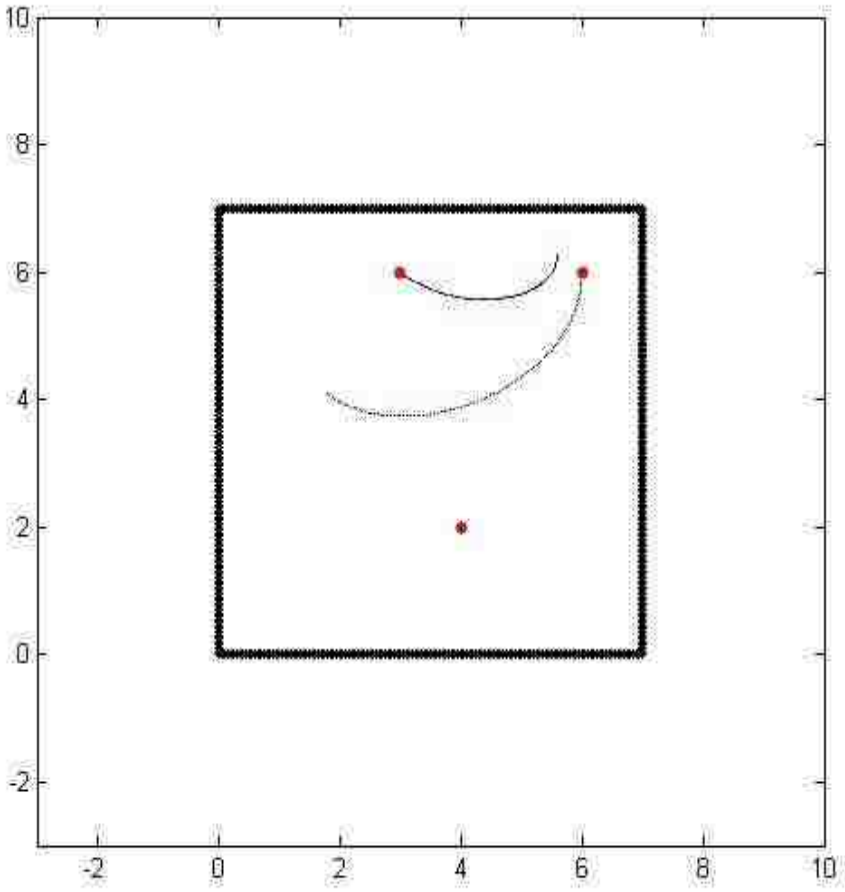


Figure 6.2. Three sensor configuration system using interpolation method.

B. Chemical particle distribution by the continuous releasing.

Particle-laden flow refers to a class of two phase fluid flow, in which one of the phase is continuously connected (referred to as the continuous or carrier phase) and the other phase is made of small, immiscible and typically dilute particles (referred to as the dispersed or particle phase). The problem of detecting odor source is typically about the particle-laden flow. The chemical particle is the dispersed phase, and the air is the carrier phase.

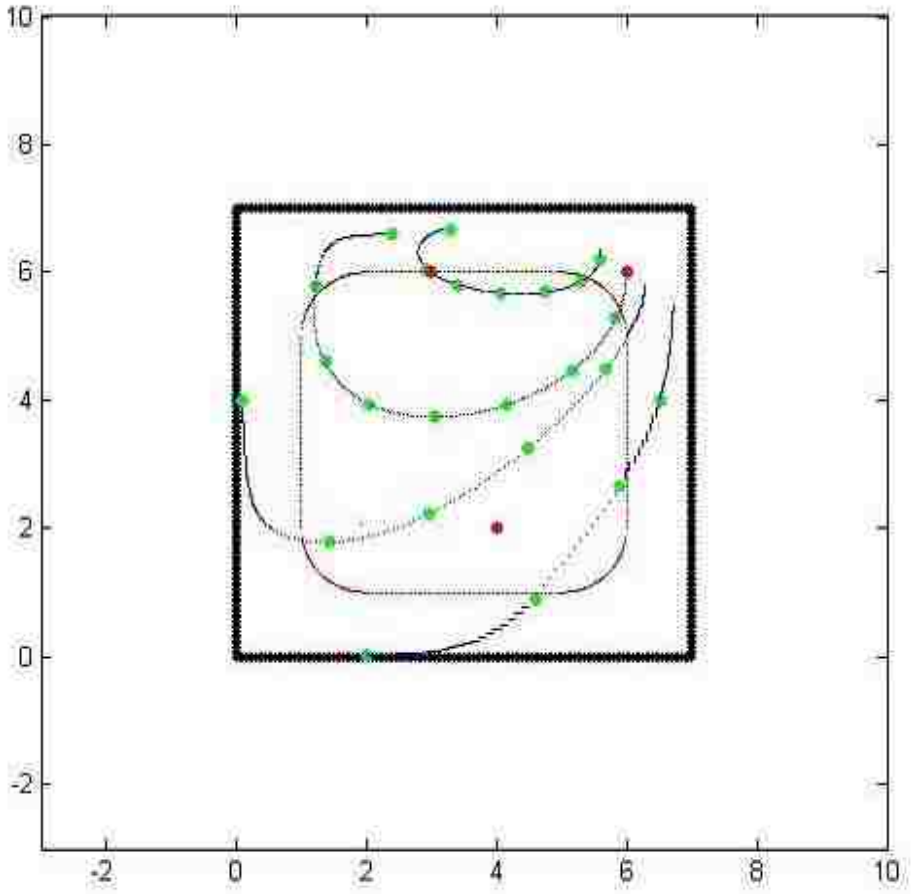


Figure 6.3. Three sensor configuration system using interpolation and extrapolation method.

If the mass fraction of the dispersed phase is small, the one-way coupling between the two phases is a reasonable assumption; that is, the dynamics of particle phases are affected by the carrier phase, but the reverse is not the case. In our case, the particles are very small and occur in low concentrations; hence the dynamics are governed by the carrier phase. The particle phase is typically treated in a Gaussian distribution along the flow direction, such that

$$C(x,y)=\frac{q}{2\pi K d_s}e^{[-\frac{u}{2K}(d_s-\Delta x)^2]},$$

where,

$$d_s=\sqrt{(x_s-x)^2+(y_s-y)^2},$$
$$\Delta x=(x_s-x)\cos\theta+(y_s-y)\sin\theta,$$

C is the concentration, q is the emitted rate, u is the wind speed, K is turbulent diffusion coefficient, θ is the angle from the x-axis to the upwind direction, and the subscript "s" denotes the odor source.

6.3. REASONING SYSTEM AND ALGORITHM

We use a reasoning system that uses the airflow model effectively to reason about the odor dispersal. It's able to navigate the sensor around the environment to gather relevant information and then successfully predict the region from which the odor originated, without moving the sensor.

The detection of odor source is finding the highest concentration in the considered area, although the author have limited number of sensors in this area. Each sensor can provide some information that contributes the decision about the location of the source.

Definition 1: When the sensor's location is (x_n, y_n), n $=1,\cdots, N$ and the odor source location is (x_s, y_s), the author use $\|(x_n, y_n)-(x_s, y_s)\|_2$ to indicate the distance. Then the closest two sensors from the minimization ($\arg\min_n \|(x_n, y_n)-(x_s, y_s)\|_2$) to the odor source, are called the critical sensors.

Definition 2: If a critical sensor is on the upstream of the chemical source, the author call it the upstream critical sensor. Otherwise, it's called the downstream critical sensor.

Through these definitions, the problem of odor source detection is transformed to the problem of detecting upstream critical and downstream critical sensors. The odor source is located in the region between the two critical sensors.

The detection process is based on the sensitivity of the interpolation with respect to individual sensors. In a system with N sensors, the author first generate a set of particle paths based on all of the sensors. Then, the author successively reduce an individual sensor data one at a time and generate another set of particle paths. The differences

between these two sets of particle paths provide us the necessary information to identify and locate the source.

To demonstrate the reasoning process, the author assume there are 4 sensors in the room, as shown in Figure 6.4. Based on the method described in Section 2, the author conclude that the airflow is in from left to right direction. In other words, the particle paths go through Sensor 1 first, then Sensor 2 and 3, and lastly Sensor 4.

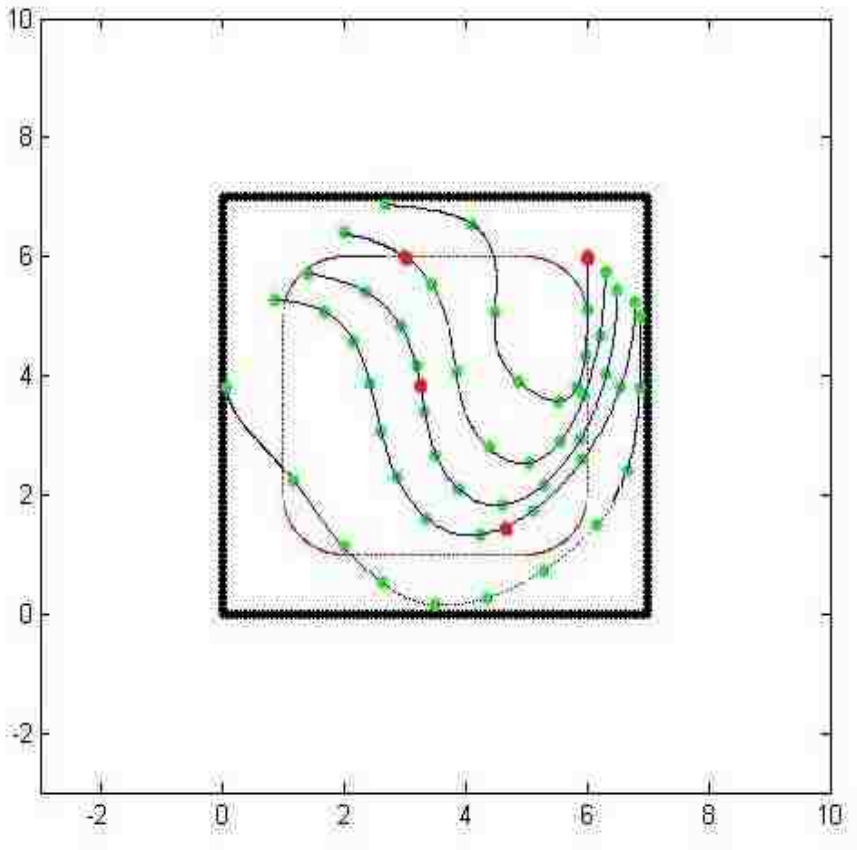


Figure 6.4. The particle path map using 4 sensors.

As part of the method, the author can approximate the particle paths, the position, the velocity, and the concentration of every point on the particle paths. Figure 6.5 shows the concentration distribution along the particle path for this case. The horizontal axis denotes the motion distance of the particles along the path, and the vertical axis shows the value of the chemical concentrations.

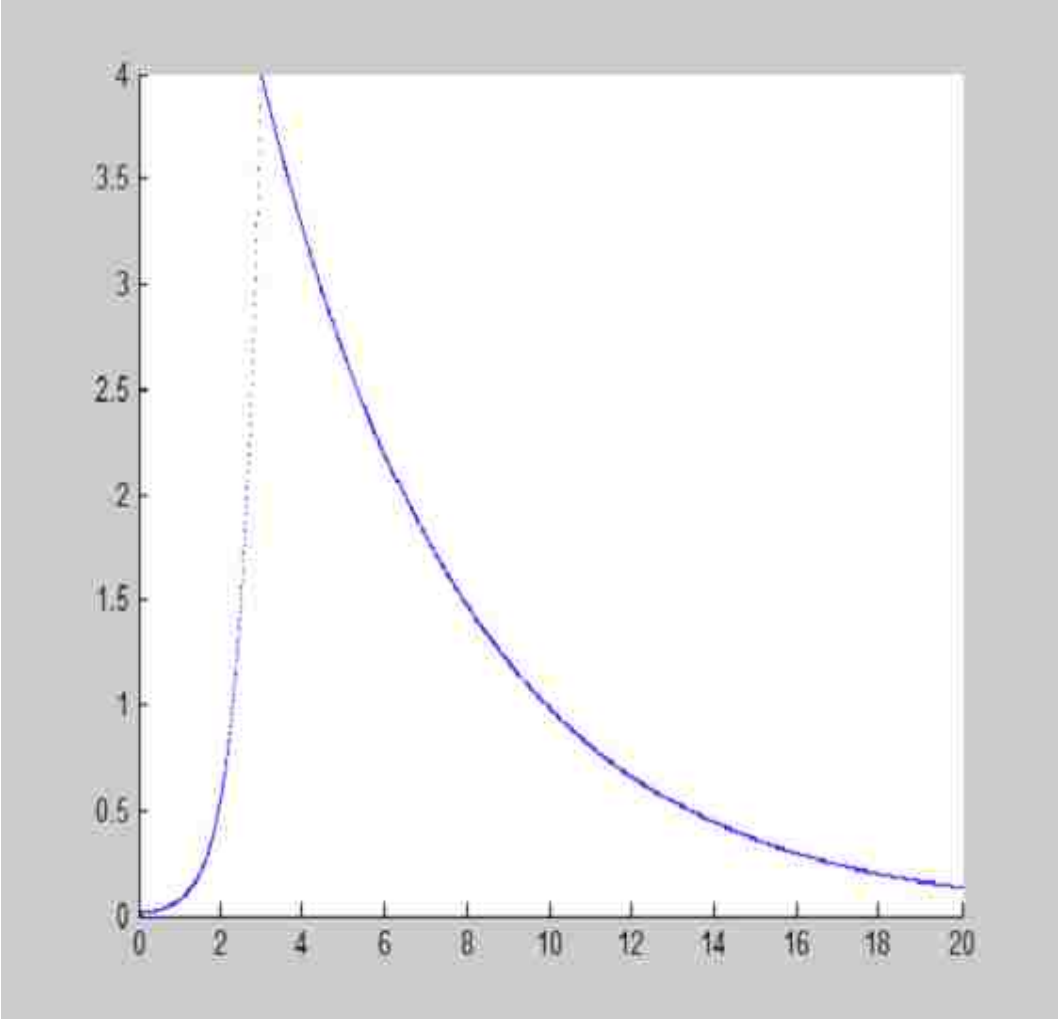


Figure 6.5. The chemical concentration on the particle path.

The odor source is located between Sensor 1 and Sensor 2. In downstream flow, the chemical concentration is decayed smoothly with a small rate, but in the upstream, the chemical concentration is decayed drastically, because the air flow blows most of particles downstream. In the example case, when the author remove Sensor 1, the updated chemical concentration at the location of Sensor 1 is higher than the original value. The author observe this result in Figure 6.6.

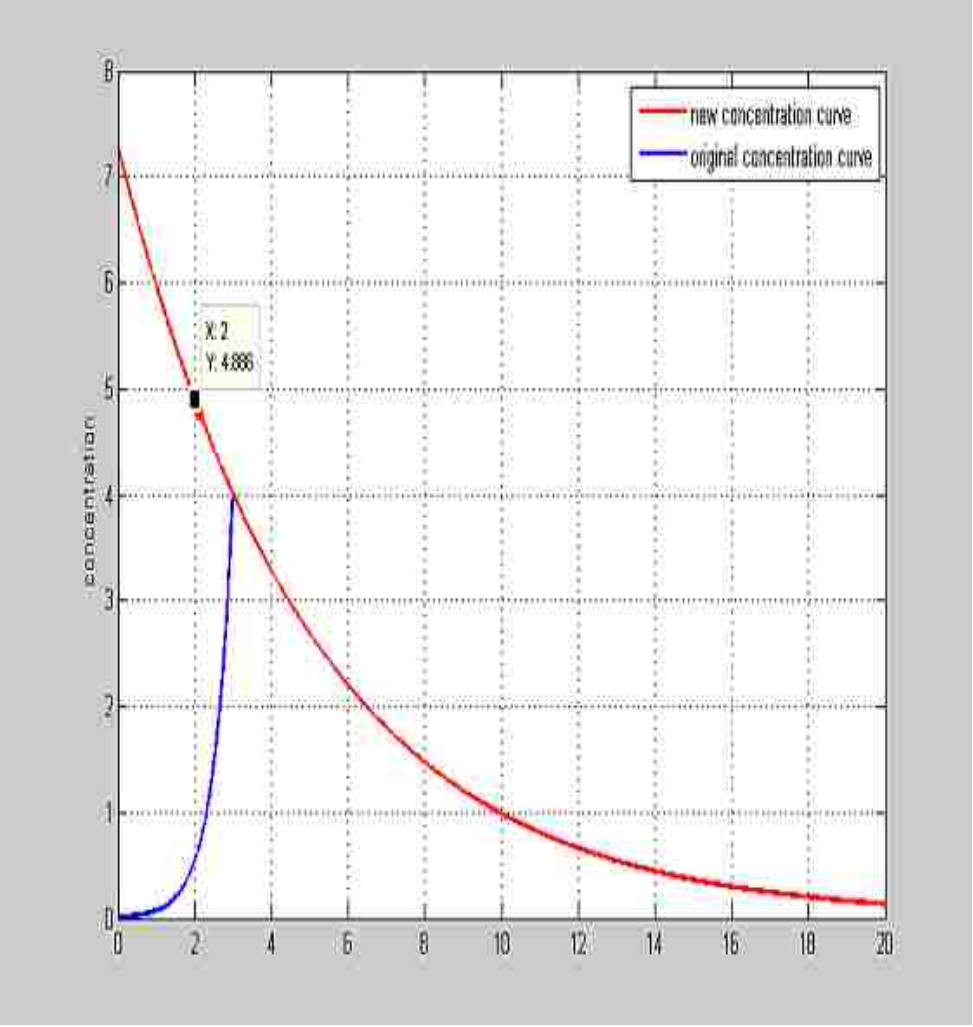


Figure 6.6. Concentration curves without critical sensors.

As a result, the author conclude that Sensor 1 is an upstream critical sensor. Applying same reasoning on Sensor 2, the author observe that the chemical concentration at the location of Sensor 2 is lower than the original value, as seen in Figure 6.7.

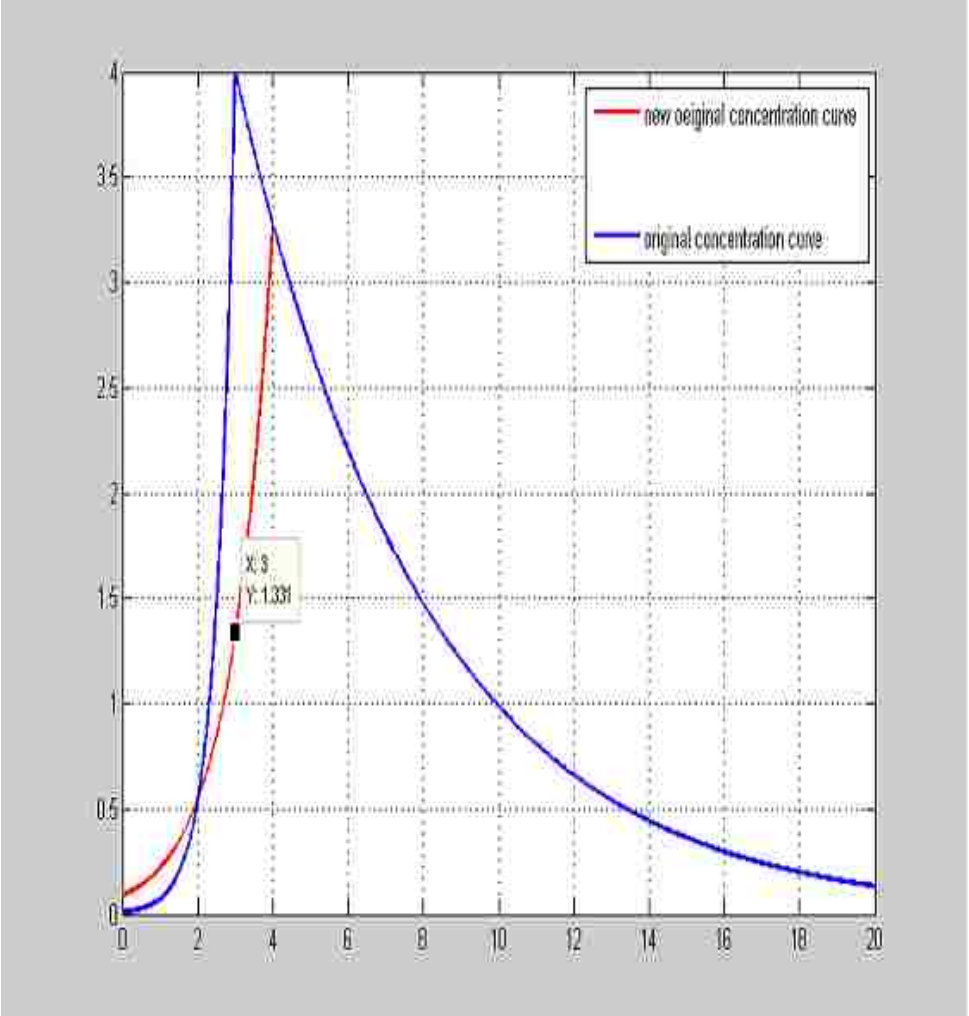


Figure 6.7. Concentration curves with critical sensors.

As a result, the author conclude that Sensor 2 is a downstream critical sensor. Similarly applying same method on Sensor 3 and Sensor 4, the author observe that the chemical concentrations at the locations of Sensor 3 and Sensor 4 are almost equal to the original values. Consequentially, the author conclude that Sensor 3 and Sensor 4 are not close to the source and they are not critical sensors. From the above analysis, the author conclude that the odor source should be located between Sensor 1 and Sensor 2.

The accuracy in the odor source detection is directly related to the amount of sensors and the placement of the sensors. Since the concentration on an upstream of the odor source cannot decrease more than a known rate, the author get a large error, when the concentration on the upstream critical sensor is higher than the concentration on the downstream critical sensor. If the value of the upstream critical sensor is larger than the value of the downstream critical sensor, then the author conclude that the source is located further upstream of the upstream critical sensor. As a result, the author can choose a wrong region as the odor source in such circumstances.

Case 1: ($S_n > S_0$ or $S_n < S_0$ case). After removing one sensor, the author get a new particle and a new chemical dispersal map. If the new chemical concentration S_n on at the location of the removed sensor is higher (or lower) than the actual valve S_0, then the author conclude that the removed sensor is upstream (or downstream) of the odor source. In this case, the removed sensor is called critical sensor.

Case 2: ($S_n \approx S_0$ case). After removing one sensor, the author get a new particle and a new chemical dispersal map. If the new chemical concentration (S_n) at the location of the removed sensor point is close to the actual valve (S_0), then the author conclude that the removed sensor is far from the odor source, and this sensor is not a critical sensor.

In the above analysis, the author concluded that the source is in the region between Sensor 1 and Sensor 2 as shown in Figure 6.8. In most cases, the author need to improve the detection by reducing the region. To achieve this reduction, the author utilize the secondary paths as described in the previous section.

Similar to the primary path approach, the author generate consistent chemical concentration at the points on the perpendicular lines to the paths going through the

critical sensors. We, then, compare these concentrations and identities the two paths with the highest concentrations as the critical paths. Figure 6.9 shows how the region that the odor source is located is narrowed using the secondary path analysis.

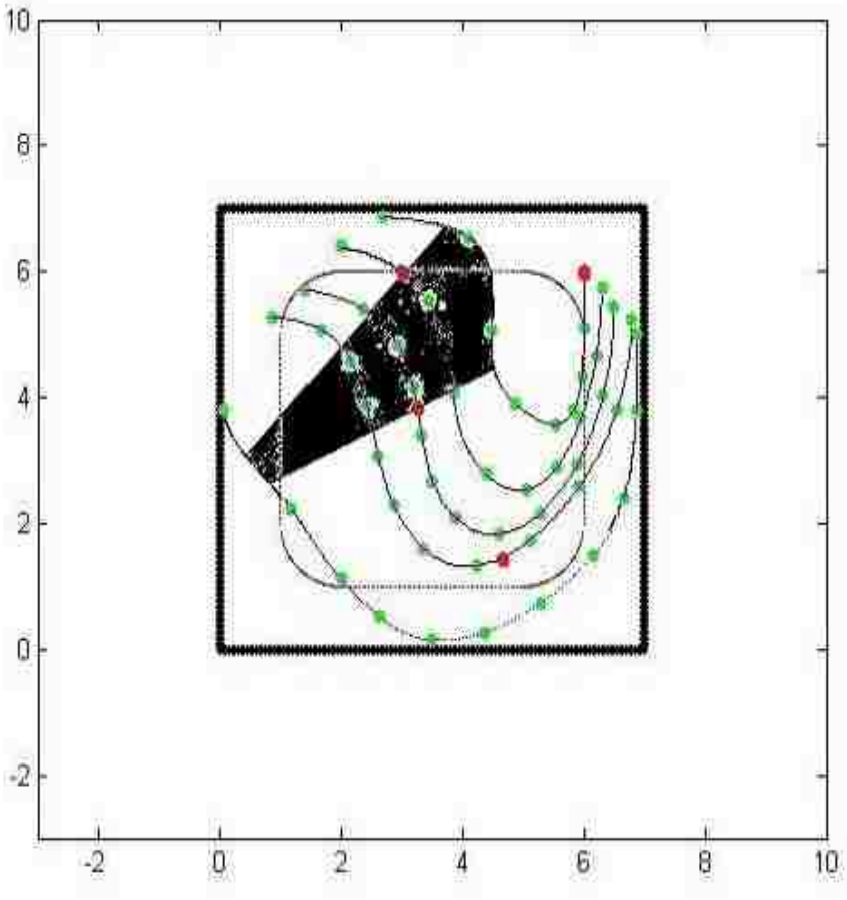


Figure 6.8. The region selected by critical sensors.

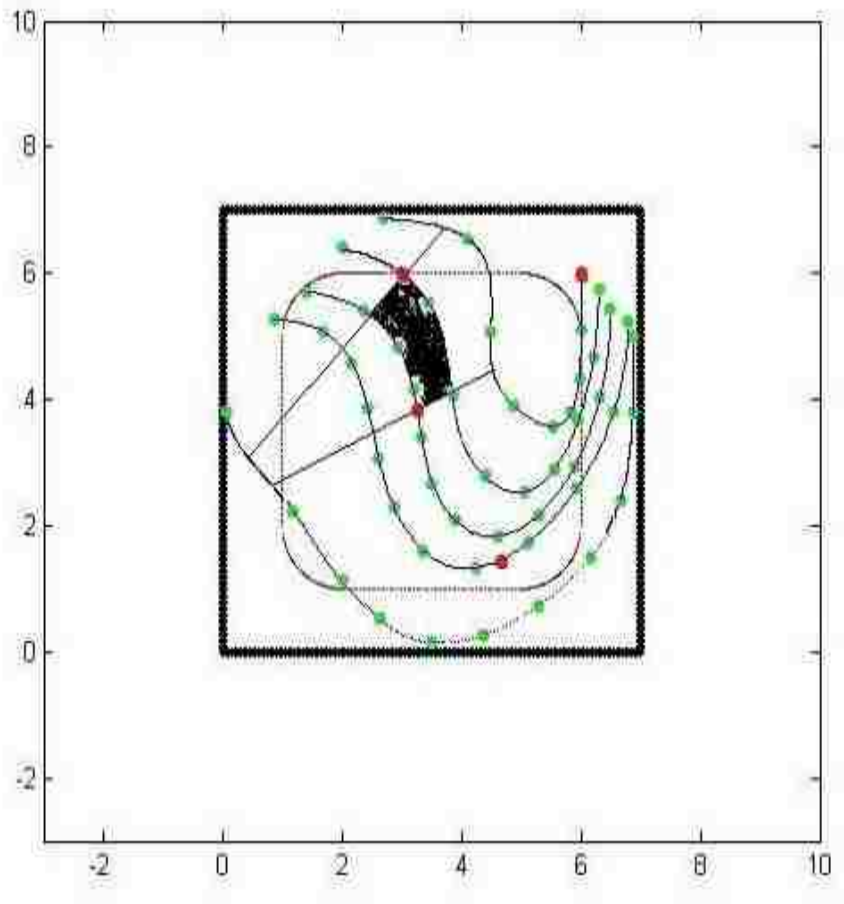


Figure 6.9. The most-likely region selected by critical sensors.

6.4. EXPERIMENTAL EVALUATIONS

In this section, the author apply the method presented on the previous section to a real world problem.

First, the author obtained a real map of Missouri University of Science and Technology campus. Second, the author use an edge detection technology to process the map to eliminate all the features except the main buildings. Figure 6.10 shows the real map after the edge detection process. Third, the author place 8 sensors on the surveyed region and generated the primary paths as shown in Figure 6.11.

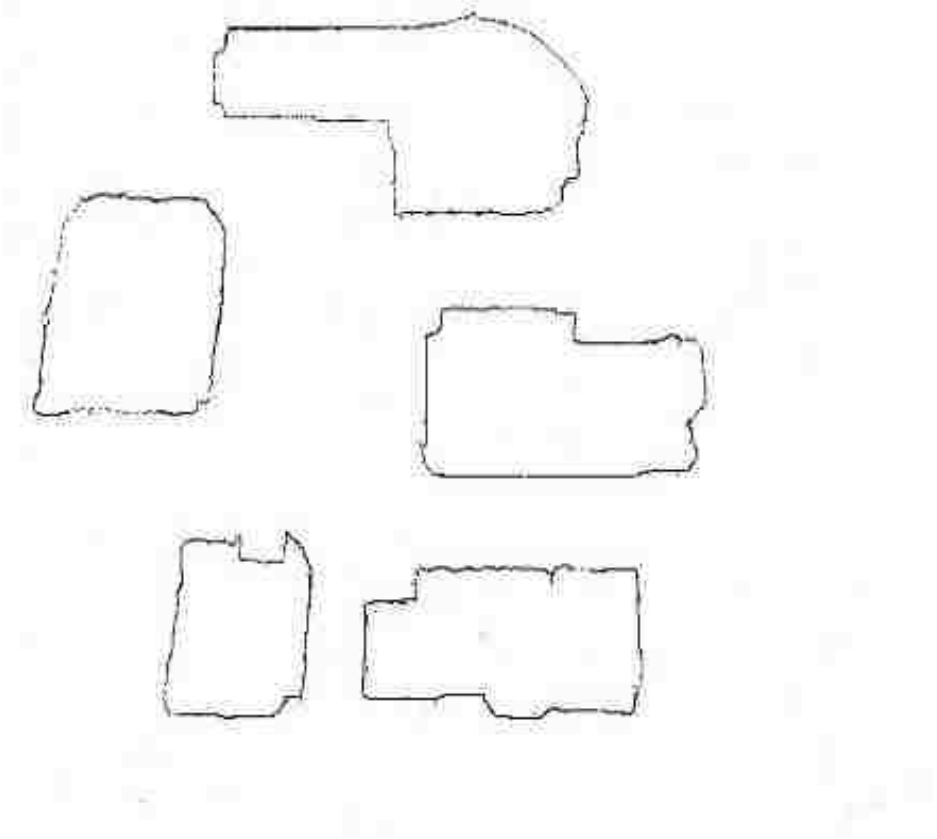


Figure 6.10. The real map processed by edge detection method.

As the author explained in the previous sections, the author removed the data of every sensor one at a time and determined the critical sensors. Based on the critical sensor data and the secondary path analysis, the author obtained the region for the source of the odor particles as shown in Figure 6.12. The black region is the most likely hood of chemical source existed.

For comparison purposes, the author also used fluid dynamics simulation to study the airflow characteristics in the same environment. The author used the COMSOL software that is used to analyze complex flow of fluid dynamics. The author set the wind to flow from southwest to northeast and the configuration is set to be the same with the

case that is solved using this dissertation's approach. The COMSOL software utilizes the finite element method that incorporates the fluid dynamics of the air flow, and plots the airflow and chemical concentration distribution.

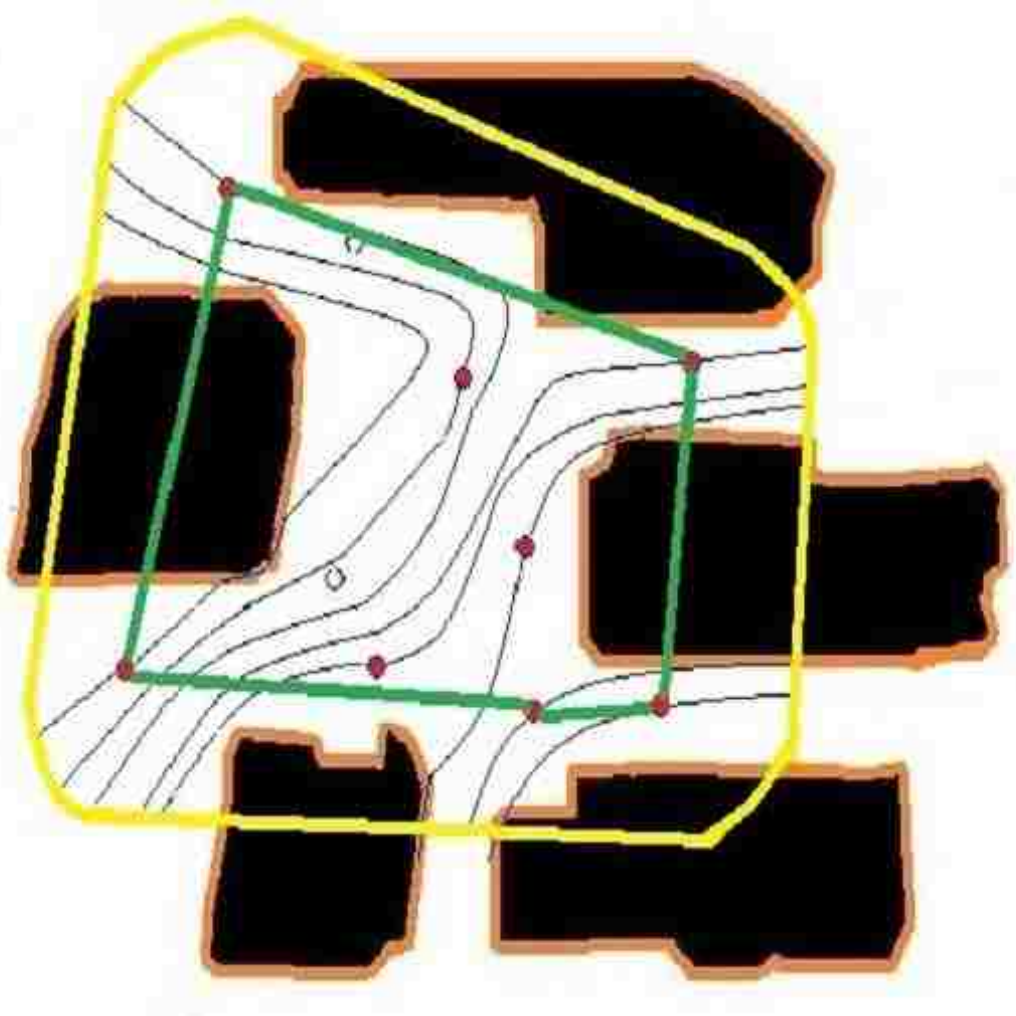


Figure 6.11. A particle paths map of Missouri University of Science and Technology.

Figure 6.13 shows the steam lines of airflow as produced by the COMSOL software. Comparing the results, the author verify that the most-probable region that contains the odor source determined by the proposed method is consistent with the COMSOL software results.

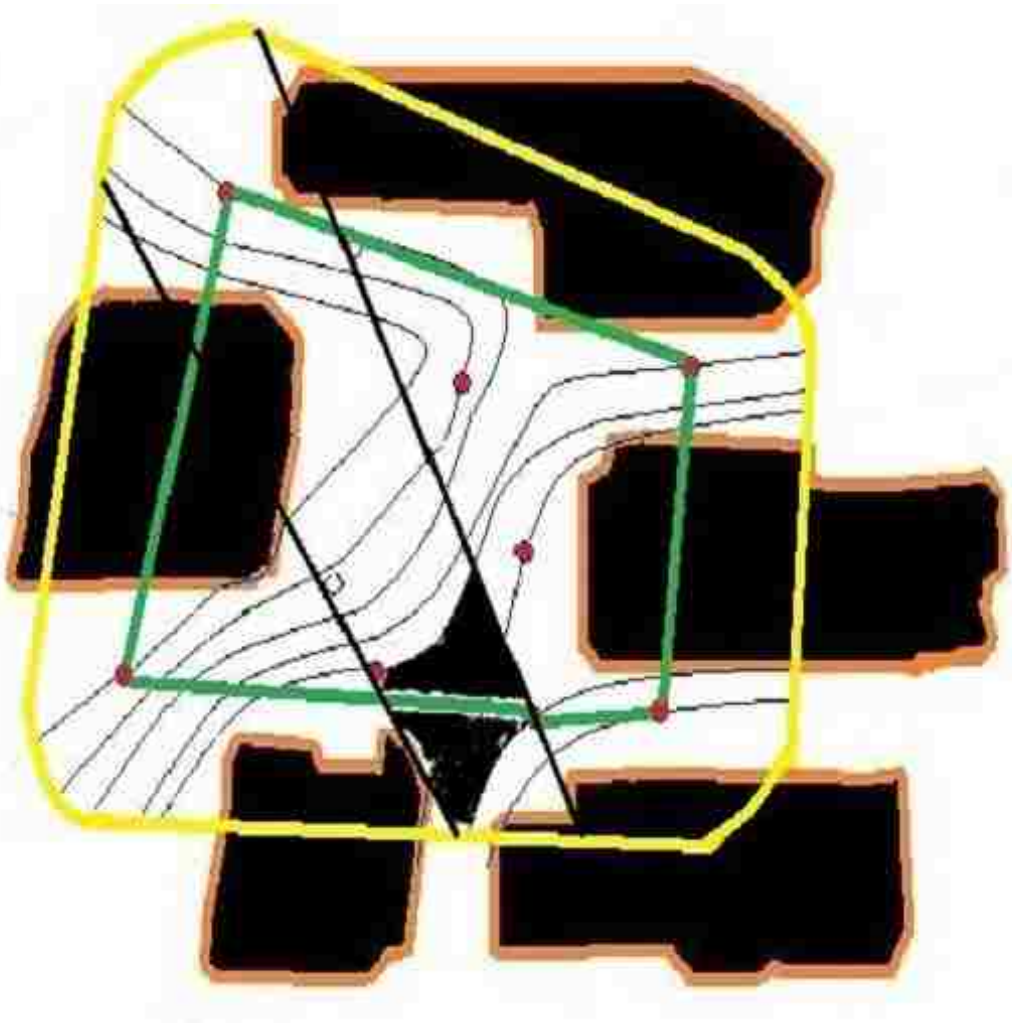


Figure 6.12. The most-likely region contains odor source in the real map.

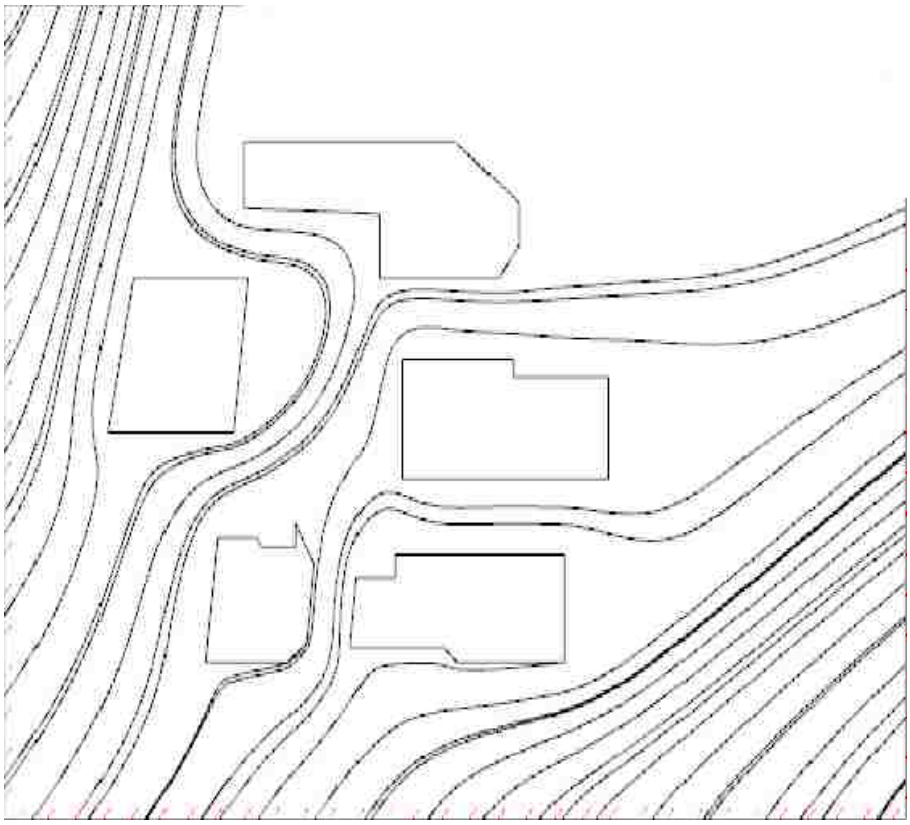


Figure 6.13. The steam lines of airflow as produced by the COMSOL software.

When the author compare the particle flow paths in Figure 12 and the air flow paths in Figure 13, the author verify the close consistency of the presented interpolation method, even though the interpolation method requires and uses at least a couple of magnitude less computational and storage resources than COMSOL software.

6.5. THE FRAMEWORK AND ALGORITHM OF THE UNMANNED AIRCRAFT CONTROL ARCHITECTURE

The important parts of a mobile robot for detecting and tracking chemical source are odor sensors and anemometer sensors. The odor distribution changes rapidly in dynamic advection-diffusion airflow environments. Therefore, odor cannot be intercepted in most workspaces, and even when the robot intercepts the odor, the interception time is brief because of the rapidly changing plume. In such a case, the robot needs to respond

immediately to the presence of the odor arrival. The odor sensor detects the presence of an airborne substance through a change in the electrical resistance of a chemically sensitive carbon-doped polymer film. It is well known that both the response time and the recovery time of the commonly used metal-oxide-semiconductor gas sensors are lengthy (of the order of several seconds to tens of seconds). Therefore, with a fixed concentration threshold in natural outdoor airflow environments, both odor detection and no detection events can lag significantly. The larger the fixed threshold, the further the odor-detection event lags behind the real odor interception, and the bigger the risk of not detecting the odor contact. For example, if the real odor concentration is 20 ppm, with the fixed threshold of 25 ppm, the robot fails to find the odor plume. The smaller the fixed threshold, the greater the chance of invalid detection events (i.e., even though the gas sensor is not in the odor plume, odor detection events still occur). To make the robot quickly and reliably respond to odor interception, a binary concentration with an adaptive threshold, rather than a fixed threshold, is used in the present study. The adaptive threshold used in this research was proposed and can be represented as:

$$\bar{S}_K = \begin{cases} \lambda \bar{S}_{K-1} + (1-\lambda)\mathrm{S}_K & K \geq 1 \\ \mathrm{S}_K & K = 0 \end{cases}$$

Where, S_K is the measured odor concentration at time step k. S_0 is the initial concentration detected at the start and $\lambda \in [0, 1]$. The binary concentration with adaptive threshold is defined as:

$$z_k = \begin{cases} 1 & S_K - \bar{S}_{K-1} > 0 \\ 0 & else \end{cases}$$

,where z_k indicates an odor detection event at time step k.

An anemometer mounted on the robot can provide the relative velocity between the airflow and the anemometer. The air-flow velocity described in this paper was a modified value, with the velocity of the robot's motion being subtracted from the anemometer measurement. The anemometer is a Shibaura F6201-1 air flow sensor, which can sense wind flow with an accuracy of 0.05 m/s. It is enclosed in a tube to provide unidirectional sensitivity which, combined with a scanning behavior, allows the robot to measure wind direction. When wind direction information is required, the robot first

rotates 90, and then rotates slowly 360 while reading the wind sensor output and finally rotates back to the heading corresponding to the highest sensor value.

In the localization of an odor source, the author implement a multi-stage process. In the first stage, the author scatter a number of sensors at nonspecific locations, perhaps by an aircraft at an area of interest, after processing the information from the sensors, the author decide to place more sensors at narrower specific locations to for the explore and focus the localization process.

First, the author decide that the area of interest or distribution and the number of sensors to be scattered. Obviously, these decision are based on the needs and availability. With the initial decision made, the author scatter the sensors within the area of distribution in a nonspecific way. This nonspecific or arbitrary locations of the initial set of sensors can be changed if the author have a priori knowledge about the odor location based some other data. Second, the author obtain data from these sensors and process the data to generate a narrower region of the source location.in the results presented here, the author have 2 sensors as the initial number of sensors. When the author obtain a narrower region than the original, the author need to decide whether or not there is really an odor source. This decision is usually easy based on the detected sensor values and the sensor sensitivity values. Third, the author scatter more sensors within the narrower region to get a refined odor particle data and to obtain an even narrow region for the location of the odor source. Finally, when the author repeat this process of scattering more sensors in a narrower and narrower area and processing more and more sensor data or when the author run out of sensors, the author moved to a small enough region where the author identify as the location of the odor source.

During the process of scattering more and more sensor into a narrower and narrower areas, either the author can choose an arbitrary and nonspecific location very similar to the initial choice or the author can direct a vehicle such as an aircraft or a robot to place the sensors at specific locations. The processing of the sensor data to narrow the area of interest for the location of the odor source can be done analytically as well as adaptively. The author explored the analytical analysis approach to narrow the area of interest. In this document, the author use this adaptive approach based on the above

method. The adaptive approach guide the system to be close to the region where it has the maximum probability.

In this paper, the author proposed a novel control algorithm of mobile robot for detecting and tracking chemical source. The block diagram in Figure 6.14 illustrates the framework of unmanned aircraft control. At each time step, the location of the odor source is estimated and the termination condition is checked. If the terminated condition is satisfied, the mission of detecting the chemical source is ended; otherwise, the unmanned aircraft moves to more locations to obtain more measurements and update the gas distribution map and odor source localization.

This tracing strategy is described as follows. At first, the unmanned aircraft moves arbitral and scatters sensors in two locations, because the chemical particle flow is derived by at least two sensor nodes. From the first step the author know roughly the source location from the gas distribution map. Then, the unmanned aircraft moves to a new location where it is along the direction of the chemical particle propagation. When the author obtain a new location and update the gas distribution map, the author have two methods. The author can use all of the sensors to map the chemical particle propagation, or the author can use a new sensor to update the map derived by the former sensors. The unmanned aircraft motion step can be adaptive through the real-time gas distribution map.

There are many useful and humanitarian reasons to locate the source of a chemical odor source. Generally, the majority of work in this area uses reactive control schemes that track an odor plume along its entire length. This type of an approach is slow and difficult in cluttered environments. In this paper, the author presented an interpolation and extrapolation method to model odor generating particle flow in an environment with distributed sensors. The author used particle paths of the model to narrow down the location of the odor source. The presented method has the advantage of utilizing at least couple of magnitude less resource than a finite element based commercial software analysis.

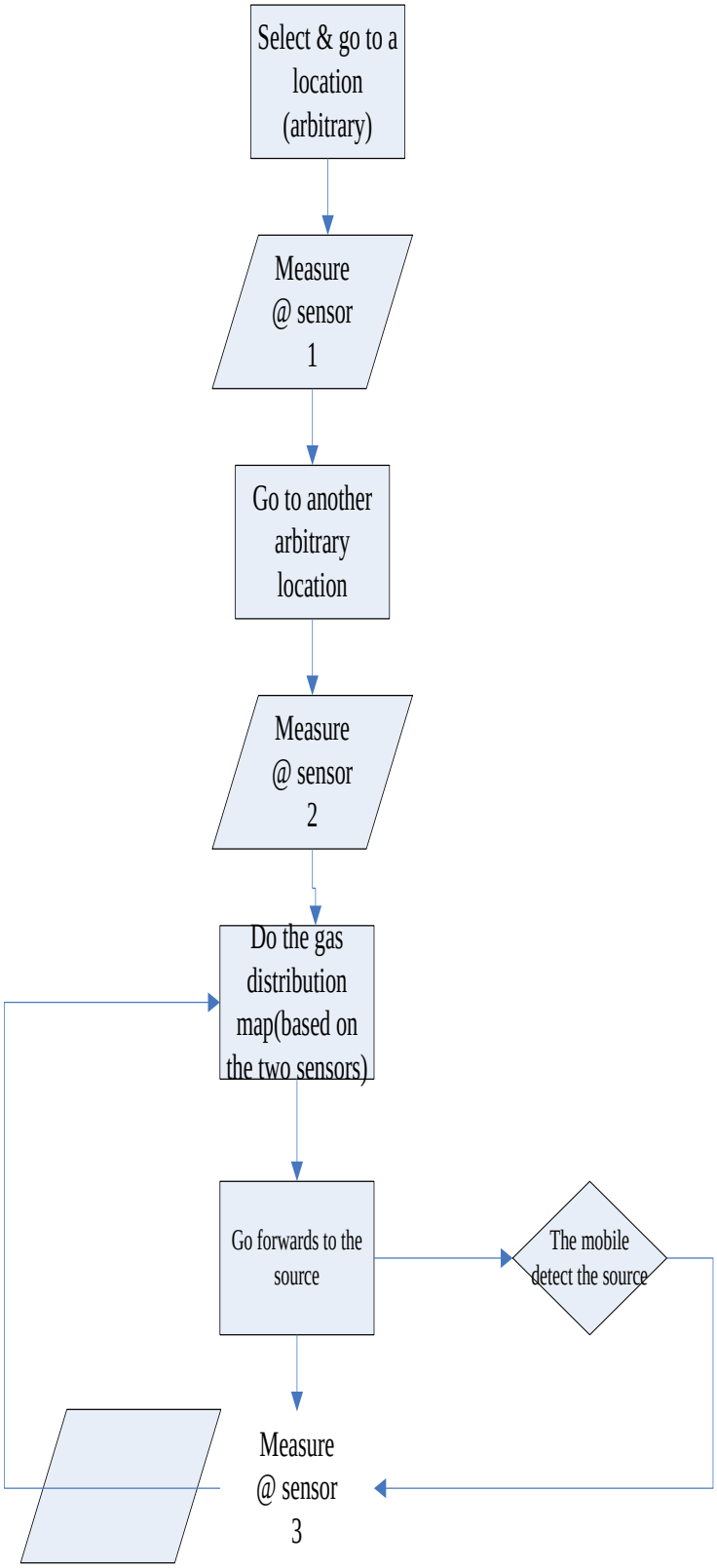


Figure 6.14. Unmanned aircraft control architecture.

6.6. RBF NEURAL NETWORK AND SOURCE DETECTION

The detection of odor source is finding the highest concentration in the considered area, although the author cannot measure everywhere in the considered area. Using limited information collected from sensors to estimate the entire area's propagation information becomes the reasoning system's core. Each sensor can provide some information that can make a decision where is the source. And considering all of the sensor's information together can get an accurate decision. It's likewise sensor mergence.

Definition 1: When the sensor's location is (x_n, y_n), n $= 1 \cdots N$ and the source's location is (x_s, y_s), the author use $\|(x_n, y_n) - (x_s, y_s)\|_2$ to indicate the distance. Then the closest ones ($\arg\min \|(x_n, y_n) - (x_s, y_s)\|_2$) to the odor source, they are named the critical sensors.

Definition 2: If the critical sensor is on the upstream of the chemical source, it is the upstream critical sensor. Otherwise, it's the downstream critical sensor.

Through these definitions, the problem of odor source detection is transformed to the problem of detecting upstream critical and downstream critical sensors. The odor source is located in the region between the two critical sensors.

In this paper, the author use neural network to detect the chemical source. In the other word, this problem is about classification. Using neural network one can classify the critical sensors and common sensors.

In the field of mathematical modeling, a radial basis function network is an artificial neural network that uses radial basis functions as activation functions. The output of the network is a linear combination of radial basis functions of the inputs and neuron parameters. Radial basis function networks have many uses, including function approximation, time series prediction, classification, and system control. They were first formulated in a 1988 paper by Broom head and Lowe, both researchers at the Royal Signals and Radar Establishment. RBF networks are typically trained by a two-step algorithm. In the first step, the center vectors of the RBF functions in the hidden layer are chosen. This step can be performed in several ways: centers can be randomly sampled from some set of examples, or they can be determined using k-means clustering. Note that this step is unsupervised. A third backpropagation step can be performed to fine-tune

all of the RBF net's parameters. The second step simply fits a linear model with weights coefficients to the hidden layer's outputs with respect to some objective function. A common objective function, at least for regression/function estimation, is the least squares function.

In the Figure 6.15, the author use the RBF network architecture which has 3 layers. Architecture of a radial basis function network.

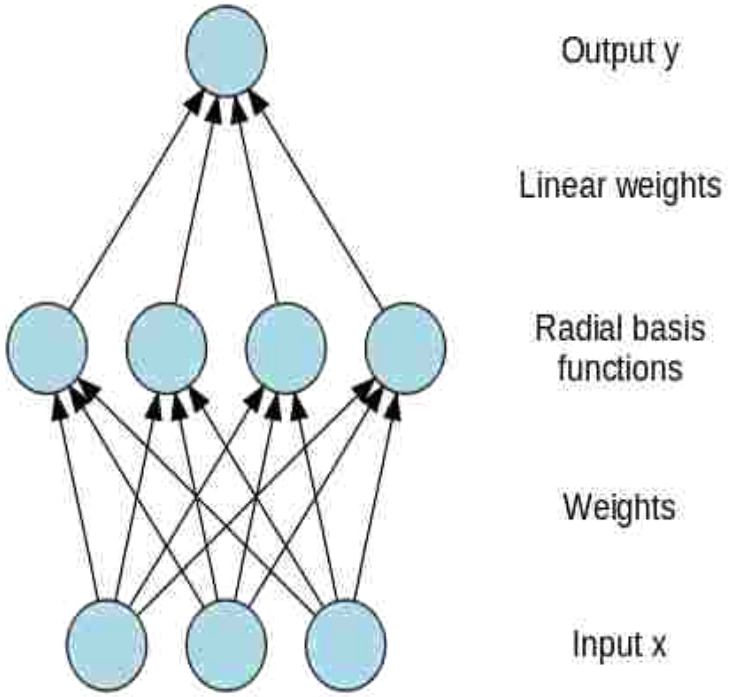


Figure 6.15. The structure of 3-layer RBF neural network.

An input vector x is used as input to all radial basis functions, each with different parameters. The output of the network is a linear combination of the outputs from radial basis functions. The number of neurons in the hidden layer is optional, but it should be

more than the inputs. The number of input is the number of sensor, which indicates every input is related with the sensors' measurement. The output is the classification result. Use Training data can teach the neural network ability to detect critical sensors. The activity function for the output is

$\varphi(x) = \sum_{i=1}^{N} a_i \rho(\|x - c_i\|)$, where ρ is a radial basis function.

Here x is a vector of the measurements by sensors, and $\mathrm{x} = (\mathrm{x}_1, x_2, \cdots x_n) \in R^n$.

The output of the neural network is

$$\mathrm{y} = \sum_{i=1}^{h} w_i \exp(-\frac{1}{2\sigma^2}\|x - c_i\|).$$

In this neural network, the author use 0 indicates the sensors are far-way sensors, 1 indicates the sensors are critical sensors.

RBF network are typically trained by a two-step algorithm. In the first step, the center vectors c_i of the RBF functions in the hidden layer are chosen. The second step simply fits a linear model with coefficients w_i to the hidden layer's outputs with respect to some objective function. A common objective function, at least for regression/function estimation, is the least square function: $K(\mathrm{w}) = \sum_{t=1}^{\infty} [\mathrm{y(t)} - \varphi(\mathrm{x(t)}, \mathrm{w})]^2$.

The input vector is the n-dimensional vector that you are trying to classify. The entire input vector is shown to each of the RBF neurons.

A Radial Basis Function Network (RBFN) is a particular type of neural network. In this article, I'll be describing its use as a non-linear classifier.

Generally, when people talk about neural networks or "Artificial Neural Networks" they are referring to the Multilayer Perceptron (MLP). Each neuron in an MLP takes the weighted sum of its input values. That is, each input value is multiplied by a coefficient, and the results are all summed together. A single MLP neuron is a simple linear classifier, but complex non-linear classifiers can be built by combining these neurons into a network.

The RBFN approach is more intuitive than the MLP. An RBFN performs classification by measuring the input's similarity to examples from the training set. Each

RBFN neuron stores a “prototype”, which is just one of the examples from the training set. When the author want to classify a new input, each neuron computes the Euclidean distance between the input and its prototype. Roughly speaking, if the input more closely resembles the class A prototypes than the class B prototypes, it is classified as class A.

Each RBF neuron stores a “prototype” vector which is just one of the vectors from the training set. Each RBF neuron compares the input vector to its prototype, and outputs a value between 0 and 1 which is a measure of similarity. If the input is equal to the prototype, then the output of that RBF neuron will be 1. As the distance between the input and prototype grows, the response falls off exponentially towards 0. The shape of the RBF neuron’s response is a bell curve, as illustrated in the network architecture diagram. The neuron’s response value is also called its “activation” value. The prototype vector is also often called the neuron’s “center”, since it’s the value at the center of the bell curve. The output of the network consists of a set of nodes, one per category that the author are trying to classify. Each output node computes a sort of score for the associated category. Typically, a classification decision is made by assigning the input to the category with the highest score.

The score is computed by taking a weighted sum of the activation values from every RBF neuron. By weighted sum the author mean that an output node associates a weight value with each of the RBF neurons, and multiplies the neuron’s activation by this weight before adding it to the total response.

Because each output node is computing the score for a different category, every output node has its own set of weights. The output node will typically give a positive weight to the RBF neurons that belong to its category, and a negative weight to the others.

The inputs are the locations, velocities, concentrations, and concentration derivatives. And the output shows the two sensors are critical or not. Figure 6.16 shows the training results. It can be seen that the network is well trained for all of sample data. After network training, the network is then verified. The author prepared three groups of the smell data: the first group for the critical sensors, the other two groups for far-way sensors. The author input the three untrained data into the neural network.

Experiments presented here were obtained using seven sensors in a room where a point sensor is set up. There are other methods that detect the critical sensors by the rules. The neural network is trained by the sample data to study the rules in the weights of neural network.

In our experiment, 10 groups of sample data are from COMSOL and 3 groups of data to check the result of classification, and in the Table 6.1 the sample data are listed. In the Table 6.1, there are seven sensors that mean 6 links are existed.

Table 6.1. Sample data from the sensors to train the RBF neural network.

x_0	y_0	s_0	δs_0	x_1	y_1	s_1	δs_1	output
1.0	2.2	1.45	-0.2	2.3	2.3	1.55	-0.2	0
2.3	2.3	1.55	-0.2	2.5	2.9	1.69	-0.4	0
2.5	2.9	1.69	-0.4	3.0	3.3	2.40	-0.8	0
3.0	3.3	2.40	-0.8	3.3	3.9	2.99	-1.2	0
3.3	3.9	2.99	-1.2	3.6	4.2	10.1	-6.2	1
3.6	4.2	10.1	-6.2	3.9	4.5	4.6	4.2	0
3.9	4.5	4.6	4.2	4.5	5.1	3.9	3.6	0
4.5	5.1	3.9	3.6	5.1	5.8	3.2	3.5	0
5.1	5.8	3.2	3.5	6.0	6.0	2.2	2.9	0
6.0	6.0	2.2	2.9	6.4	6.8	2.0	2.8	0

The inputs are the locations, velocities, concentrations, and concentration derivatives. And the output shows the two sensors are critical or not. Figure 6.16 shows the training results. It can be seen that the network is well trained for all of sample data.

After network training, the network is then verified. The author prepared three groups of the smell data: the first group for the critical sensors, the other two groups for far-way sensors. The author input the three untrained data into the neural network.

The Table 6.1 shows the neural network's inputs and outputs. This neural network has eight inputs and 1 outputs. The inputs are the values that are collected by the sensors, while the output is the sensor label. The label indicates the sensor is a critical sensor or a far-away sensor. Using data from historical sensors, the neural network is trained to learn the ability to distinguish the critical sensors.

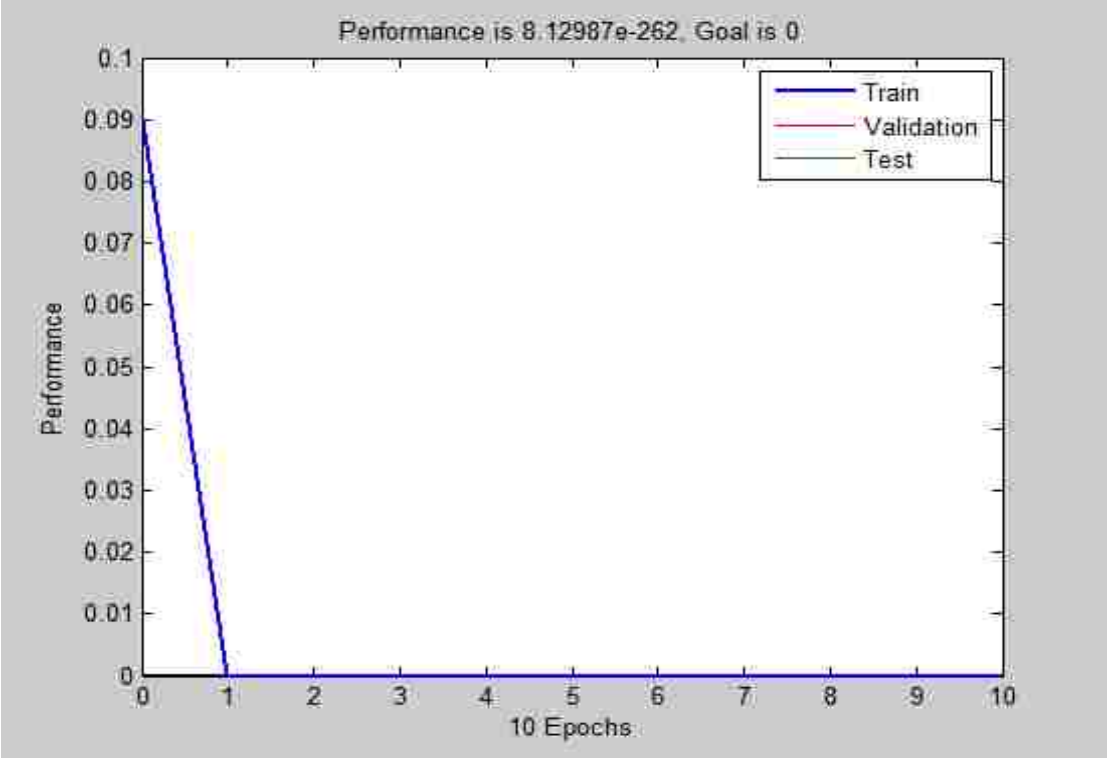


Figure 6.16. The performance of RBF neural network.

Table 6.2 shows the detection results for the three groups of testing data using RBF neural network. Therefore, the neural network is proved to have the possibility of quick detection of the nearest (critical) sensors to the chemical source. From the result, the author can see the odor source should be located between the sensors (3.4, 3.8) and (3.7, 4.1).

Table 6.2. Untrained sample data for the neural network.

x_0	y_0	s_0	δs_0	x_1	y_1	s_1	δs_1	output
2.5	2.7	1.68	-0.4	3.1	3.1	2.35	-0.7	0
3.4	3.8	3.00	-1.3	3.7	4.1	10.2	-6.3	1
5.2	4.2	3.9	3.7	6.1	6.0	2.2	2.9	0

6.7. CONCLUSION

There are many useful and humanitarian applications that can locate the source of a chemical source. Currently, the majority of work in this area uses reactive control schemes that track an odor plume along its entire length, which is slow and difficult in cluttered environments. This paper employs a high-level control scheme. The interpolation and extrapolation method is used to model the particle path in the sensors' environment. Then a reasoning system use the path model to get the velocity, chemical concentration at any point on the map and predict the most probable locations of the odor source. This approach has been shown to be effective for odor localization in a known environment, without the need for the robot to travel to the source.

This paper presented an odor detection of chemical source by using the functions of neural network in training and data recognition based on smell data. In order to make a quick detection, neural network was employed. The author trained the network with the

positions of sensors of the smell data of the odor. The trained network was then tested with the untrained data. The experiment results show that the neural network is well trained and could acquire distinguishing ability for the critical sensors or far-way sensors.

7. CONCLUSION

Chemical source localization and detection is very worthy research in various research. In the military, this research can help human being to work in dangerous environments, track the explosive substances, like bombs. In the environment engineering, it can be used to find the air pollutant in the open-air fields by deploying the drones and wireless sensor networks. In the public security, this research can detect smoke in the forests, and search for survivors in the relics. These are just a few applications of this research. There are more other areas where this research is very useful.

In the dissertation, the author make a lot of contributions about chemical propagation mapping, chemical concentration distributions, sensor fusion, and chemical source localization, unmanned vehicle's application.

The interpolation and extrapolation are proposed to model the air flow between two different positions. Using chemical concentration value, the model from the interpolation and extrapolation method can match at the two end points. After the link between two points, the author can use this model to implement the links among all of the nodes. The nodes and the links compose the chemical particle paths. In addition, this dissertation give some rules to model the chemical particle paths by going through the opening in the area of obstacles and blocking in the region of obstacles. These rules also can be the prerequisites for the collision-free algorithms. At last, along the particle paths, the chemical particle distributed by Gussian distribution. By then, the author can get the chemical concentration and the airflow paths, so that the chemical particle propagation mapping has been finished.

The above method has very import usage, because it save computational time and memories. Traditional methods have much more complex calculation to get the chemical propagation mapping. Even the sampling based method doesn't have to use chemical propagation mapping, it still need to process tons of real-time data to detect chemical source by updating the uncertainty at every iteration. However, the results between the traditional methods and our novel method are very close. Validation by computational fluid dynamics indicates our model has the same level of accuracy and efficiency. Simple

cases that there are simple airflow and environment models are calculated by hands. Complex cases are simulated by the business software COMSOL that is embedded with finite element method. In addition. In our simulations and experiments, the author consider in many factors, the obstacles, the openings, the numbers of sensors, the dispatchment of sensors and the outdoor environment.

This dissertation has good contributions to sensor fusion technology. When more sensors are scattered on the area of interest, the more data are gotten. Using multi-senor networks to get better accuracy of the chemical propagation mapping is a challenge in the recent research. The author give two kinds of method to smooth the chemical particle paths. The new sensors only affect the paths around the sensors. It's obvious that the new sensors are local data that cannot change the whole area of the interest. So, the updated parameters of cubic spline functions is the main idea of smooth the chemical particle paths.

Detecting chemical source is completed by our reasoning system and control scheme. The author propose direct method and indirect method. The direct method can be used in the new environment. By the reasoning system and control scheme, the author can induce the maximum likely hood of chemical source. Around the area near the chemical source, the chemical concentration cannot increase and decrease continuously. The chemical source always make the chemical concentration dramatically changed. So, if the sensors that are closest to the chemical source can by classified, the chemical source will be detected. The indirect method is applied to the environment where the previous date has been collected. A lot of previous data can be used to train the RBF neural network weights. Then, the network can classify the sensors are close to the chemical source or the sensors are far with the source. When the author get the critical sensors, the chemical source are navigated in a narrower area where the vertex are the critical sensors.

Recently, more and more people and more and more new technology join in the research about chemical source detection. Optical sensors, Radar sensor and ultrasonic sensors can be used in this research. Because most of the above sensors are non-contact technologies, they have more applications for chemical source detection, especially for some dangerous or explosive substances. Integration with the different type of sensors is

a challenge to get better analysis of the data. Using extend Kalman filter and Particle filter can integrate the data from different sensors to update the chemical concentration, chemical particle path and vehicle navigation. The other challenge is the cooperation of multi-robot. Multi-robot is a more complex problem. It need to think about the share information among different robots and collision-free algorithms among different robots. The author think the author can achieve the artificial intelligence in the soon future, as long as the author make progress step by step.

BIBLIOGRAPHY

[1] Lili, N. A. "Accuracy of Determination for Virtual Animal Halal Slaughtering: A Case Study," 2010 Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation. IEEE, 2010.

[2] Peiyuan, Guo, and Bao Man. "Pattern recognition to lamb freshness based on much data merging," 2011 IEEE International Conference on Mechatronics and Automation. 2011.

[3] Ajemba, P., et al. "A low cost smell detection system for multimedia applications," Consumer Electronics, 2003. ICCE. 2003 IEEE International Conference on. IEEE, 2003.

[4] Loutfi, Amy, and Silvia Coradeschi. "Odor recognition for intelligent systems," IEEE intelligent Systems 23.1 (2008): 41-48.

[5] Langer, K. "A guide to sensor design for land mine detection," The Detection of Abandoned Land Mines: A Humanitarian Imperative Seeking a Technical Solution, EUREL International Conference on (Conf. Publ. No. 431). IET, 1996.

[6] Charumporn, Bancha, et al. "Early stage fire detection using reliable metal oxide gas sensors and artificial neural networks," Neural Networks, 2003. Proceedings of the International Joint Conference on. Vol. 4. IEEE, 2003.

[7] Bona, Evandro, et al. "Optimized neural network for instant coffee classification through an electronic nose," International Journal of Food Engineering 7.6 (2011).

[8] Baltes, Henry. "Micro and nano sensors snoop around," MEMS, NANO and Smart Systems, 2003. Proceedings. International Conference on. IEEE, 2003.

[9] Tudu, Bipan, et al. "Smell peak prediction during black tea fermentation process using time-delay neural network on electronic nose data," Computing: Theory and Applications, 2007. ICCTA'07. International Conference on. IEEE, 2007.

[10] Ishida, Hiroshi, et al. "Controlling a gas/odor plume-tracking robot based on transient responses of gas sensors," IEEE Sensors Journal 5.3 (2005): 537-545.

[11] Castro, Rafael, et al. "An electronic nose for multimedia applications," IEEE Transactions on Consumer Electronics 49.4 (2003): 1431-1437.

[12] Khomh, Foutse, et al. "A bayesian approach for the detection of code and design smells," 2009 Ninth International Conference on Quality Software. IEEE, 2009.

[13] Kodogiannis, Vassilis S., et al. "Artificial odor discrimination system using electronic nose and neural networks for the identification of urinary tract infection," IEEE Transactions on Information Technology in Biomedicine 12.6 (2008): 707-713.

[14] Hassaine, Salima, et al. "IDS: an immune-inspired approach for the detection of software design smells," Quality of Information and Communications Technology (QUATIC), 2010 Seventh International Conference on the. IEEE, 2010.

[15] Kiatweerasakul, Montep, and T. John Stonham. "Odour plume tracking robot using semiconductor gas sensors," Control, Automation, Robotics and Vision, 2002. ICARCV 2002. 7th International Conference on. Vol. 2. IEEE, 2002.

[16] Kowadlo, Gideon, and R. Andrew Russell. "To naively smell as no robot has smelt before," Robotics, Automation and Mechatronics, 2004 IEEE Conference on. Vol. 2. IEEE, 2004.

[17] Nakamoto, Takamichi, et al. "Reproduction of scent and video at remote site using odor sensing system and olfactory display together with camera," Sensors, 2008 IEEE. IEEE, 2008.

[18] Munro, Matthew James. "Product metrics for automatic identification of" bad smell" design problems in java source-code," 11th IEEE International Software Metrics Symposium (METRICS'05). IEEE, 2005.

[19] Biel, Lena, and Peter Wide. "An intelligent model approach for combination of sensor information," Haptic Virtual Environments and Their Applications, IEEE International Workshop 2002 HAVE. IEEE, 2002.

[20] Steel, Travis, Dane Kuiper, and R. Z. Wenkstern. "Context-aware virtual agents in open environments," 2010 Sixth International Conference on Autonomic and Autonomous Systems. IEEE, 2010.

[21] Lozowski, Andrzej G., Mykola Lysetskiy, and Jacek M. Zurada. "Signal processing with temporal sequences in olfactory systems," IEEE Transactions on Neural Networks 15.5 (2004): 1268-1275.

[22] Xu, Dongming, and José Carlos Principe. "Dynamical analysis of neural oscillators in an olfactory cortex model," IEEE Transactions on Neural Networks 15.5 (2004): 1053-1062.

[23] Allen, Jacob N., et al. "An E-nose haar wavelet preprocessing circuit for spiking neural network classification," 2008 IEEE International Symposium on Circuits and Systems. IEEE, 2008.

[24] Polá, I. "A visual based framework for the model refactoring techniques," Applied Machine Intelligence and Informatics (SAMI), 2010 IEEE 8th International Symposium on. IEEE, 2010.

[25] Gao, Xiang, and Levent Acar. "Using a mobile robot with interpolation and extrapolation method for chemical source localization in dynamic advection-diffusion environment," IAES International Journal of Robotics and Automation (IJRA) 5.2 (2016).

[26] Gao, Xiang, and Levent Acar. "Multi-Sensor Integration to Map Odor Distribution for the Detection of Chemical Sources," Sensors 16.7 (2016): 1034.

[27] Gao, Xiang, and Levent Acar. "Detection and Tracking of an Odor Source in Sensor Network Using a Reseaon System," Journal of Automation and Control Engineering Vol.4, No. 6, 2016.

[28] Gao, Xiang, and Levent Acar. "Multi-Sensor Integration to Map Odor Distribution for the Detection of Chemical Sources," International Journal of Technical Research & Science, Vol.1, Issue 5, 2016.

[29] Gao, Xiang, and Levent Acar. "Chemical Source Localization and Detection Using Interpolation method and the Complementary Spline Function Neural Network," International Journal of Technical Research & Science, Vol.1, Issue 5, 2016.

[30] Farrell, Jay A., Shuo Pang, and Wei Li. "Chemical plume tracing via an autonomous underwater vehicle," IEEE Journal of Oceanic Engineering 30.2 (2005): 428-442.

VITA

Xiang Gao, Bachelor of Engineering was earned at North China Electric Power University in 2007, Master of Engineering was earned at North China Electric Power University in 2011. Doctor of Philosophy (Electrical Engineering) was earned at Missouri University of Science and Technology in December 2016.