

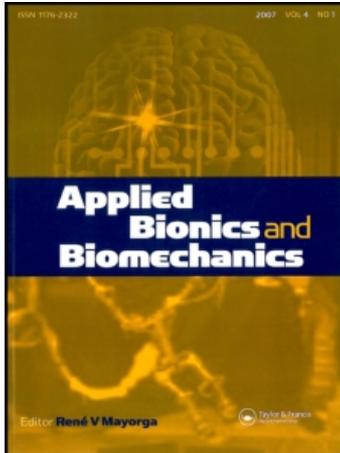
This article was downloaded by: [University of California, Los Angeles]

On: 10 March 2010

Access details: Access Details: [subscription number 918974475]

Publisher Taylor & Francis

Informa Ltd Registered in England and Wales Registered Number: 1072954 Registered office: Mortimer House, 37-41 Mortimer Street, London W1T 3JH, UK



Applied Bionics and Biomechanics

Publication details, including instructions for authors and subscription information:

<http://www.informaworld.com/smpp/title~content=t778164488>

Identifying an odour source in fluid-advected environments, algorithms abstracted from moth-inspired plume tracing strategies

Wei Li ^{ab}

^a Department of Computer Science, California State University, Bakersfield, CA, US ^b Robotics State Key Laboratory, Chinese Academy of Sciences Shenyang, Shenyang Institute of Automation, Liaoning, China

Online publication date: 01 March 2010

To cite this Article Li, Wei(2010) 'Identifying an odour source in fluid-advected environments, algorithms abstracted from moth-inspired plume tracing strategies', Applied Bionics and Biomechanics, 7: 1, 3 – 17

To link to this Article: DOI: 10.1080/11762320903537790

URL: <http://dx.doi.org/10.1080/11762320903537790>

PLEASE SCROLL DOWN FOR ARTICLE

Full terms and conditions of use: <http://www.informaworld.com/terms-and-conditions-of-access.pdf>

This article may be used for research, teaching and private study purposes. Any substantial or systematic reproduction, re-distribution, re-selling, loan or sub-licensing, systematic supply or distribution in any form to anyone is expressly forbidden.

The publisher does not give any warranty express or implied or make any representation that the contents will be complete or accurate or up to date. The accuracy of any instructions, formulae and drug doses should be independently verified with primary sources. The publisher shall not be liable for any loss, actions, claims, proceedings, demand or costs or damages whatsoever or howsoever caused arising directly or indirectly in connection with or arising out of the use of this material.

Identifying an odour source in fluid-advected environments, algorithms abstracted from moth-inspired plume tracing strategies

Wei Li*

Department of Computer Science, California State University, Bakersfield, CA US; Robotics State Key Laboratory, Shenyang Institute of Automation, Chinese Academy of Sciences Shenyang, Liaoning China

(Received 20 August 2008; final version received 7 December 2009)

This paper presents algorithms for identifying the odour source of a chemical plume with significant filament intermittency and meander developed in fluid-advected environments. The algorithms are abstracted from moth-inspired chemical plume tracing strategies in two steps. First, we introduce the concept of the last chemical detection points that leads to construction of a source identification zone and development of two variations in the source identification algorithms. Second, we use Monte Carlo methods to optimise the proposed algorithms in a simulated environment. The evaluation results demonstrate that the optimised algorithm achieves a success rate of over 90% in identifying the source location, the average identification time is 3–4 min and the average error is 1–2 m surrounding the source location.

Keywords: biologically inspired searching strategy; chemical plume tracing; odour source localization; olfaction-based navigation

1. Introduction

Recently, there has been a growing interest to apply a robot-based chemical plume tracer in homeland security and environmental monitoring. In order to develop the plume tracer for a natural fluid environment application, DARPA/ONR sponsored an interdisciplinary team to investigate biologically inspired chemical plume tracing (CPT) strategies under the CPT program (Cowen and Ward 2002). Grasso et al. (2000) implemented biomimetic strategies on their robot lobster to analyse chemotaxis-based algorithms and to evaluate biomimetic strategies. Grasso and Atema (2002) employed a single sensor or two sensors detecting fluorescence to compare the plume-tracing performance of three plume-tracing strategy variations. Other studies (Liao and Cowen 2002; Weissburg et al. 2002) proposed sensor array-based strategies and suggested that search strategies based on following the ‘edge’ of a plume (as opposed to the centreline are robust). Hayes et al. (2002) used multiple robots to improve a Spiral Surge Algorithm in the field of swarm intelligence in order to find a plume and to trace the plume to its source location. Li et al. (2001) developed, evaluated and optimised both *passive* and *active* plume tracing strategies inspired by moth behaviour. The strategies were implemented on a REMUS underwater vehicle with a single chemical sensor for the in-water test runs in November and April 2002 at the San Clemente Island of California (Figure 1) and in June 2003 in Duck, North Carolina (Farrell et al. 2005; Li et al. 2006). The field experiments success-

fully demonstrated tracking of chemical plumes over 100 m and source identification on the order of tens of metres in the near shore, oceanic fluid flow environments, where plumes were developed under turbulence, tides and waves. Nevertheless, the source identification still leaves significant room for improvement.

This paper systemically discusses a process of designing source identification algorithms, which are abstracted from the moth-inspired plume tracing strategies based on a single chemical sensor (Li et al. 2001). First, we introduce the concept of the last chemical detection points (LCDPs) to construct source identification zones (SIZs) (Li 2006) and use chemical detection events to develop two variations of SIZ algorithms for the source identification along with the measured tracer locations and fluid flow directions. The first algorithm, which we call SIZ_T, maintains the constant number of the most recent LCDPs in a priority queue in the time sequence. This algorithm computes the minimum and maximum coordinates of the LCDPs in the queue to construct an SIZ, and dynamically monitors the SIZ_T size during CPT missions. When the SIZ_T size becomes small enough, it identifies the mean of the LCDPs as the source location. The second algorithm named SIZ_F keeps *all* the detected LCDPs in the order of the current up-flow direction. A SIZ_F holds a constant size. This algorithm dynamically checks the number of LCDPs inside SIZ_F during CPT missions. When the SIZ_F contains enough LCDPs, it identifies the most up-flow LCDP inside SIZ_F as the odour

*Email: wli@csusb.edu

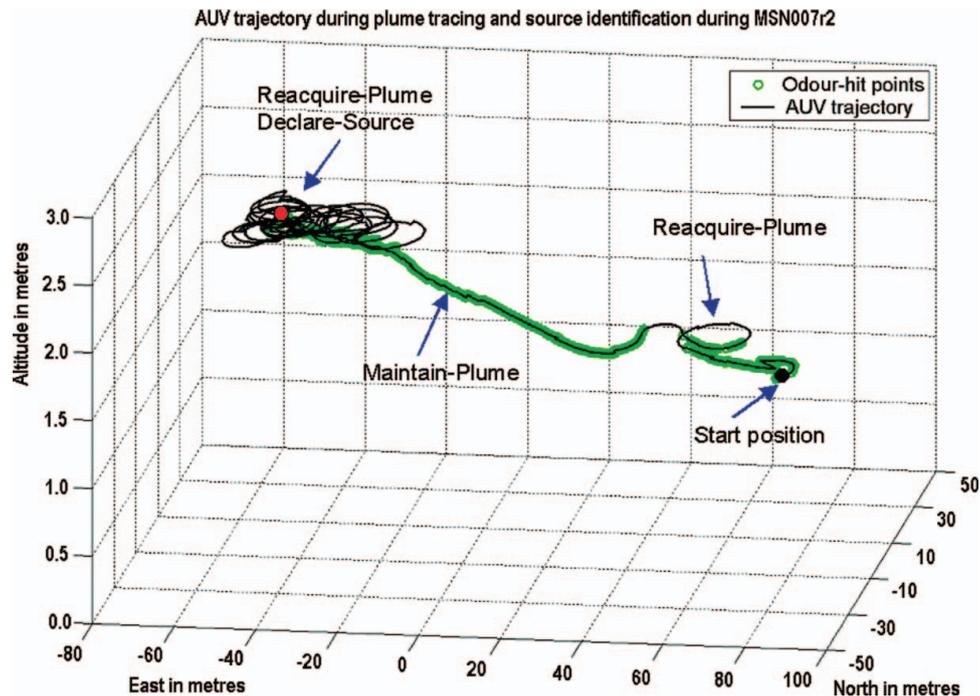


Figure 1. A CPT in-water test run conducted in November 2002 at San Clemente Island, California, using a plume of Rhodamine dye developed in near shore ocean conditions. REMUS vehicle trajectory was documented over distance of 411 m from the first detection point to the identified source location for the time in [480.3, 1190.6] s. The grey circles indicate locations where the chemical concentration was above threshold when the vehicle was at that location.

source location. Second, we adopt Monte Carlo methods to optimise the algorithms using a simulated plume with significant filament intermittency and meander (Farrell et al. 2002), and evaluate the performance of source identification manoeuvres in three aspects: reliability, identification time, and accuracy.

This paper is organised as follows. In section 2, we state the source identification problem. In section 3, we define the concept of the last chemical detection points (LCDPs) and present the two SIZ algorithms. In section 4, we analyse the proposed algorithms. In section 5, we develop a strategy to evaluate and optimise the algorithms. In section 6, we draw conclusions about the algorithms.

2. Problem statements of source identification

Several groups have studied olfactory-based navigation in structured environments. Russell (2001) included robotic implementation of algorithms that estimate statistics of the plume such as the plume centroid and experiments where the chemical is constrained to a multiple-duct tunnel system. Ishida et al. (1996, 2001) used an array of sensors to track the plume by estimating the plume centreline, by estimating parameters of a Gaussian plume-distribution model or by estimating the three-dimensional (3-D) direction toward the odour source. Marques et al. (2002) performed plume tracing tests using mobile robots in laboratory en-

vironments. Lilienthal et al. (2004) reported test results on proximity to a gas source when a laboratory robot manoeuvred in the vicinity of the source location in an environment with scales of centimetres. However, the assumed conditions in the experiments are hardly available in the real world, and there exist multiple local concentration maxima when a plume is propagated in a turbulence-dominated flow environment. Recently, Li (2009) used six robots to localize an odour source in a laboratory environment, but the approaches still need to be validated for tracing a plume in a natural environment.

For real application, the objective of source declaration is to identify the chemical source in a natural environment, where a robot-based tracer must be able to trace a chemical plume over a significant distance to its source location. Factors that complicate the source identification include the unknown chemical source concentration, different advection distance of any detected chemical and the significant flow variation with both location and time. Different from plume tracing, there is no clear analogue to the robot Declare-Source behaviour for animals. For biological entities (e.g., moths), the conclusion of identifying the pheromone source location may still be a mystery. Instead, while moth plume tracing relies primarily on sensed pheromones, the final determination of the location of the female moth could be based on data from multiple sensors, including vision, tactile or even auditory cues. In this study,

however, available information for making source identification includes a set of above threshold concentrations detected by a single chemical sensor, along with the measured instantaneous flow directions and robot manoeuvre positions.

A straightforward idea of the source identification is to estimate the odour source based on chemical detection points (odour-hit points) during plume tracing activities. The tracer has ceased progress up the plume because of inter-hit distances of the odour-hit points. Nonetheless, whether the tracer has reached the source is not foolproof because small inter-hit distances might occur in all parts of the plume (Hayes et al. 2002). The inter-hit distances are sensitive to the sampling of the control system and the predefined concentration threshold of a plume-tracing algorithm. The fluid mechanics studies show that at medium and high Reynolds numbers, the evolution of the chemical distribution in the flow is turbulence dominated (Murlis 1986; Webster et al. 2001). As a result, the turbulent diffusion process leads to a highly discontinuous and intermittent distribution of the chemical plume, which makes source identification based on odour-hit points more arduous.

3. Design of source identification algorithms

3.1. Last chemical detection point (LCDP)

We derive the source identification algorithms from the two moth-inspired behaviours: Maintain-Plume and Reacquire-Plume. Maintain-Plume is broken down into Track-In and Track-Out activities because of intermittency of a chemical plume transported in a fluid flow environment (Li et al. 2001). The tracer alternatively utilises Maintain-Plume and Reacquire-Plume in making progress towards the source location in the up-flow direction. In a typical plume tracing scenario, the tracer activates Track-In once it detects the chemical, e.g., the activities during ΔT_1 and ΔT_3 in Figure 2(a). It continues Track-Out when it loses contact with the chemical within λ seconds, e.g., the activity during ΔT_2 in Figure 2(a). After λ seconds, it switches to Reacquire-Plume for casting the plume, e.g., the activity after ΔT_4 in Figure 2(a). A chemical detection point at which the tracer loses contact with the chemical plume for λ seconds is defined as a LCDP, e.g., point $(x_{\text{last}}, y_{\text{last}})$ at T_{last} in Figure 2(a). In our applications, the coordinates of $(x_{\text{last}}, y_{\text{last}})$ are specified in a coordinate system with the origin defined by the centre of an operation area (OpArea). We use a cloverleaf trajectory with the centre $(x_{\text{last}}, y_{\text{last}})$ to implement the Reacquire-Plume behaviour, as shown in Figure 2(b). We choose the length of each leaf by considering that the minimum value is constrained to be larger than the tracer turning radius (10–15 m for the REMUS vehicle). Note that one leaf is aligned with the down-flow direction for the tracer to rediscover the chemical when it has passed the source location. During a Reacquire-Plume

activity, the tracer either detects the chemical or completes the cloverleaf trajectory N_{re} times ($N_{\text{re}} = 2$ or 3 for the in-water tests). If N_{re} repetitions are completed without a chemical detection, the tracer reverts to Find-Plume.

In order to develop the source identification algorithms, we define a LCDP node by

```

struct LCDP_Node
{
    double Tlast, xlast, ylast;
    double conc, fdir, fmag;
    double xflow, yflow;
};

```

where T_{last} is the time when the LCDP is detected, $(x_{\text{last}}, y_{\text{last}})$ are the coordinates of the tracer at T_{last} , conc is the chemical concentration at $(x_{\text{last}}, y_{\text{last}})$ and T_{last} , $(f_{\text{dir}}, f_{\text{mag}})$ are the flow direction and magnitude at $(x_{\text{last}}, y_{\text{last}})$ and T_{last} , and $(x_{\text{flow}}, y_{\text{flow}})$ are the coordinates of $(x_{\text{last}}, y_{\text{last}})$ in a new coordinate system defined according to the current flow direction. For convenience, we also use $(x_{\text{last}}, y_{\text{last}})$ to represent the LCDP in the following discussion. Note that conc and f_{mag} in the LCDP node do not appear in the source identification algorithm, but they are reserved for possible further application. In the moth-inspired CPT strategies, the chemical sensor works as a “binary detector”. The Boolean value is “1” if the chemical concentration is above the threshold. Otherwise, the Boolean value is set to “0”. The Monte Carlo study in (Li et al. 2001) shows that decreasing the threshold increases the time the tracer stays “in the plume”, but it is accompanied by an increase in noise. The threshold value was chosen as $\text{conc} > 4\%$ of the full scale (i.e., 0.2 V) on the basis of an analysis of chemical sensor data from the REMUS operating in San Diego Bay in absence of the chemical (Li et al. 2006). In this scenario, the sensor readings were pure noise but never surpassed 0.2 V. The CPT strategies adopt this threshold for both the in-water tests and the simulation studies. The proposed algorithms make the source identification decision on the basis of the number of LCDPs in SIZ instead of their concentrations. A limited increment of this threshold may affect the number of odour-hit points detected during a Maintain-Plume activity, i.e., it may increase the number of Track-Out activities. However, any Track-Out activity within λ seconds does not generate a new LCDP.

3.2. Patterns for source identification

LCDPs provide important information about plume traversal distances between Reacquire-Plume activities. The LCDPs are separated along the axis of the plume when the tracer is far from the source location, while the LCDPs are clustered in the vicinity of the source when the tracer is approaching the source location. The tracer usually exits the plume and moves up flow from the source when it traces the plume to the source location. When this situation occurs,

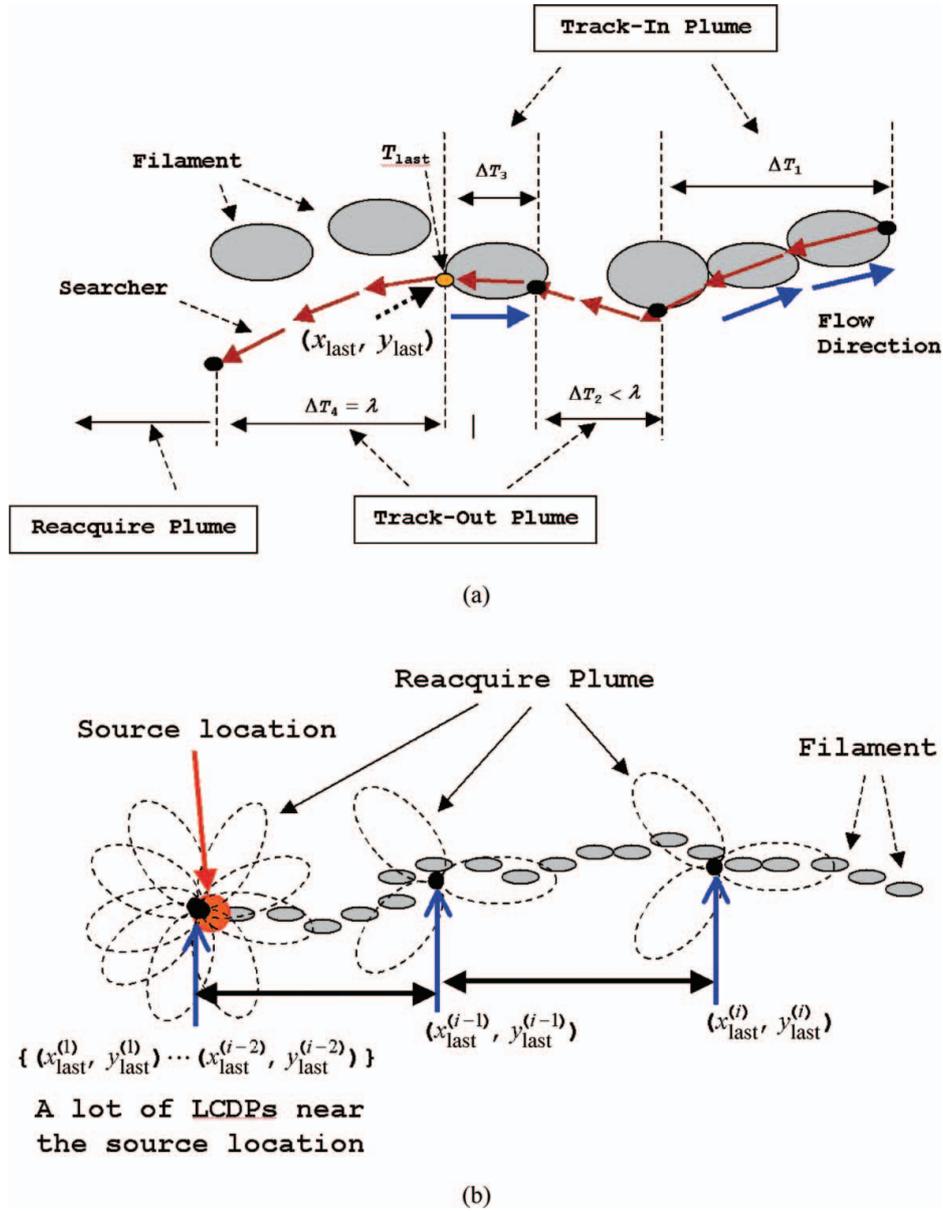


Figure 2. Derivation of source identification algorithms from moth-inspired plume tracing strategies. (a) Definition of a last chemical detection point: the tracer moves up-flow when it detects concentration above the threshold and still keeps an up-flow motion with an offset when it detects concentration below the threshold. The tracer records (x_{last}, y_{last}) as a LCDP if it cannot re-catch the plume within λ seconds during Track-Out activity. (b) Patterns for identifying the odour source using the LCDPs: the tracer generates most of the LCDPs in the vicinity of the source location, when it overshoots the source and re-catch the plume using Reacquire-Plume. Reacquire-Plume is implemented using a cloverleaf trajectory, and its centre is located at (x_{last}, y_{last}) .

the tracer also activates Reacquire-Plume to rediscover the plume on a cloverleaf trajectory. As a result of the frequent switching between Maintain-Plume and Reacquire-Plume, the tracer generates a pattern with a number of cloverleaf trajectories in the vicinity of the source location, as shown in Figure 2(b). Such a distribution of the LCDPs is employed to facilitate development of the source identification algorithm.

The tracer detects a new LCDP and inserts its node into the priority queue when it switches its behaviour from

Maintain-Plume to Reacquire-Plume. The queue sorts the LCDP nodes in a new coordinate system, defined in order of the current up-flow direction, $f_{dir} + 180^\circ$. Its x -axis is aligned with the f_{dir} direction, and its origin is located at (x_{last}, y_{last}) . The algorithm maps each LCDP in the queue into the new coordinate system by

$$\begin{bmatrix} x_{flow} \\ y_{flow} \end{bmatrix} = \begin{bmatrix} \cos(f_{dir} + 180^\circ) & -\sin(f_{dir} + 180^\circ) \\ \sin(f_{dir} + 180^\circ) & \cos(f_{dir} + 180^\circ) \end{bmatrix} \begin{bmatrix} x_{last} \\ y_{last} \end{bmatrix}. \quad (1)$$

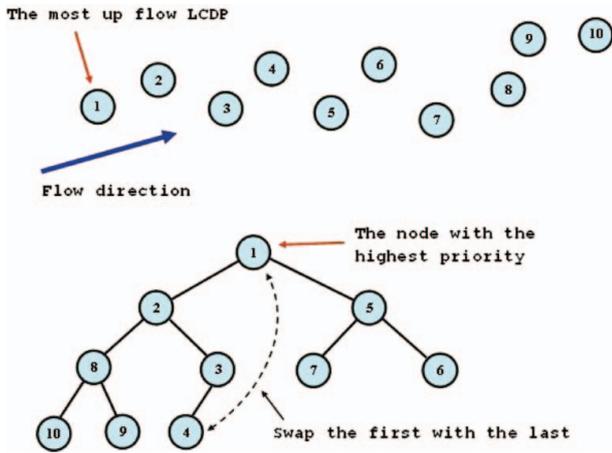


Figure 3. Maintain the LCDPs with a heap-based priority queue. The lowest number indicates the highest priority. Copy an item from the priority queue to a list needs following steps: first, the queue swaps the first LCDP node (LCDP with 1) with the last LCDP node (LCDP with 4); second, it copies the LCDP node with the highest priority to the list and deletes the last node; third, the queue compares the LCDP node (LCDP with 4) on the top with its two children, swaps with the smallest child if the LCDP node is greater than its children or child and repeats this comparison until the LCDP node is less than both of its children.

The x_{flow} components determine the LCDP nodes' priorities according to the current up flow direction. The smallest x_{flow} has the highest priority. The more LCDPs the priority queue accumulates, the more information about the source location the tracer gathers. The top panel of Figure 3 illustrates ten LCDPs in circles, and the bottom panel of Figure 3 shows that they are sorted in order of the current up flow direction. The smaller the number associated with a LCDP, the higher the priority assigned to the node.

The top node always indicates the most up flow LCDP. In each planning cycle, the identification algorithm copies the LCDPs from the queue to a list using the following procedure. (1) The queue swaps the top node (the LCDP with the highest priority) with the last node. (2) The queue copies the LCDP with the highest priority to the list and deletes the last node. (3) The queue moves a new LCDP with the highest priority to the top, as shown in the bottom panel of Figure 3. The queue repeats this procedure until it moves all the LCDPs to the list. Subsequently, the list receives the LCDPs according to their priorities.

3.3. SIZ_T algorithm

The SIZ_T algorithm keeps updating the N_{dec} most recent LCDPs during CPT missions and sorts them in the order of time serials using the priority queue. The algorithm constructs the SIZ_T size (shown in the left figure of Figure 4) by

$$\begin{aligned}
 x_{\text{last}}^{(\min)} &= \min \{x_{\text{last}}^{t(i)}\} \\
 x_{\text{last}}^{(\max)} &= \max \{x_{\text{last}}^{t(i)}\} \\
 y_{\text{last}}^{(\min)} &= \min \{y_{\text{last}}^{t(i)}\} \\
 y_{\text{last}}^{(\max)} &= \max \{y_{\text{last}}^{t(i)}\} \quad i = 1, \dots, N_{\text{dec}} \quad (2)
 \end{aligned}$$

where a superscript t indicates that the queue sorts the N_{dec} LCDPs in time series. When the tracer approaches the odour source, the distances between the LCDPs become smaller, i.e., the SIZ_T size becomes smaller. During CPT missions,

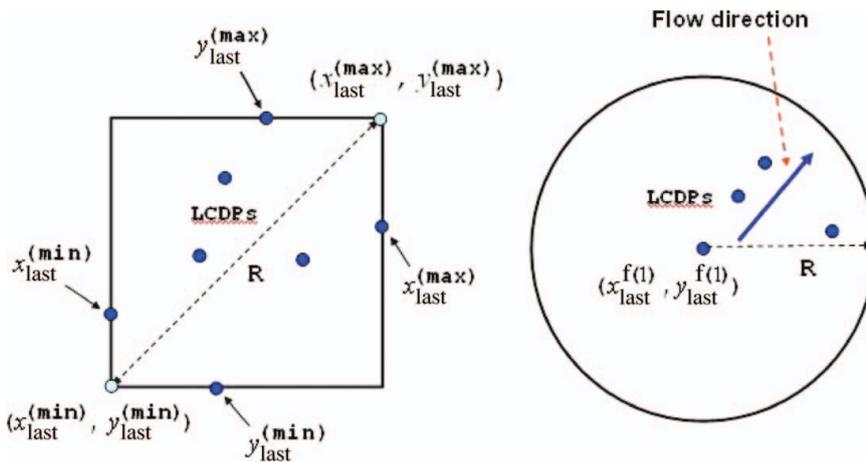


Figure 4. The left figure illustrates SIZ_T in the shape of a rectangle, which is determined by computing the minimum and maximum components of the LCDPs. SIZ_T identifies the source location when the size becomes small enough. The right figure illustrates SIZ_F in the shape of a circle. SIZ_F holds a constant size and identifies the source location when it contains enough LCPDs.

Table 1. Pseudo code for SIZ_T algorithm

ALGORITHM SIZ_T($Q[1, \dots, N_{\text{dec}}]$)
 //Identifying the source location by SIZ_T algorithm
 //Input: Priority queue $Q[1, \dots, N_{\text{dec}}]$
 //Output: Status of source identification
if ($N_{\text{dec}} \geq N_{\text{ini}}$)
 Sort Q in the order of the time serials
 for $i \leftarrow 1$ **to** N_{dec} **do**
 $P[i] \leftarrow Q[i]$ // P is a list
 Calculate $(x_{\text{last}}^{(\min)}, y_{\text{last}}^{(\min)})$ and $(x_{\text{last}}^{(\max)}, y_{\text{last}}^{(\max)})$ in Equation (2)
 Calculate the diameter of **SIZ_T** in Equation (3)
 if the diameter $\leq \varepsilon_T$
 return $(x_{\text{last}}^{(m)}, y_{\text{last}}^{(m)})$ as the source location
 else
 return no source location identified
else
 return no source location identified

the SIZ_T algorithm dynamically checks the diagonal

$$R = \sqrt{(x_{\text{last}}^{(\max)} - x_{\text{last}}^{(\min)})^2 + (y_{\text{last}}^{(\max)} - y_{\text{last}}^{(\min)})^2}. \quad (3)$$

When $R \leq \varepsilon_T$, SIZ_T identifies the mean

$$\begin{aligned} x_{\text{last}}^{(m)} &= \sum_{i=1}^{N_{\text{dec}}} x_{\text{last}}^{t(i)} / N_{\text{dec}} \\ y_{\text{last}}^{(m)} &= \sum_{i=1}^{N_{\text{dec}}} y_{\text{last}}^{t(i)} / N_{\text{dec}} \end{aligned} \quad (4)$$

as the source location. Table 1 lists the pseudo code of the SIZ_T algorithm with three parameters: the criterion, ε_T , for checking the SIZ_T size; the integer, N_{dec} , indicating the constant number of LCDPs maintained in the queue, and the initial value, N_{ini} , identifying the priority queue for accumulation of at least N_{ini} LCDPs before starting source identification. The parameters ε_T and N_{dec} are adjustable and crucial to achieving the desired performance of source identification.

3.4. SIZ_F algorithm

The SIZ_F algorithm maintains *all* LCDPs in the order of the current up flow direction using the priority queue. SIZ_F holds a constant size, ε_F , and makes the source identification with the following iterative construct: First, SIZ_F calculates $(x_{\text{last}}^{(m)}, y_{\text{last}}^{(m)})$ of all the LCDPs; Second, SIZ_F find the point, p_{max} , with the largest distance to $(x_{\text{last}}^{(m)}, y_{\text{last}}^{(m)})$

$$\begin{aligned} D_{\text{max}} &= \max \left\{ \sqrt{(x_{\text{last}}^{f(i)} - x_{\text{last}}^{(m)})^2 + (y_{\text{last}}^{f(i)} - y_{\text{last}}^{(m)})^2} \right\} \\ &\quad \times (i = 1, 2, \dots, N_{\text{all}}) \end{aligned} \quad (5)$$

from the priority queue, where a superscript f indicates that the LCDPs are sorted in the order of the most recent up flow direction and N_{all} is the total number of LCDPs detected during a CPT mission. If D_{max} is greater than ε_F , SIZ_F removes the LCPD with p_{max} from the set of LCDPs. These calculations repeat until all remaining LCDPs are close enough to $(x_{\text{last}}^{(m)}, y_{\text{last}}^{(m)})$, i.e., they all are located inside SIZ_F, as shown in the right figure of Figure 4. If the number of the remaining LCDPs is greater than N_{min} , SIZ_F identifies its most up flow LCPD as the odour source. Table 2 lists the pseudo code of the SIZ_F algorithm with three parameters: the SIZ_F size, ε_F , the initial value, N_{ini} , and the integer, N_{min} , which indicates the minimum number of LCDPs remaining inside SIZ_F for the source identification. The SIZ_F algorithm also has two the adjustable parameters ε_F and N_{min} . There are three main differences between SIZ_F and SIZ_T: (1) SIZ_F maintains all the LCDPs using the queue, while SIZ_T updates only the constant number of the most recent LCDPs in each planning cycle. (2) SIZ_T is in the shape of a rectangle, while SIZ_F is in the shape of a circle, as shown in Figure 4. (3) The SIZ_F algorithm uses an iterative construct to cluster LCDPs inside SIZ_F, while SIZ_T uses a sequential construct to check its size. The parameter, N_{ini} , defined in the algorithms works as a filter to block some invalid LCPDs only when N_{dec} or N_{min} is very small.

4. Algorithms analysis

4.1. Simulated fluid-advected plume

We evaluate the SIZ algorithms using a tracer with the REMUS dynamics in a simulated flow environment (Farrell et al. 2002), as shown in Figure 5. The simulated plume model achieves significant computational simplification relative to turbulence models, but it was designed to

Table 2. Pseudo code for SIZ_F algorithm

```

ALGORITHM SIZ_F( $Q[1, \dots, N_{\text{all}}]$ )
//Identifying the source location by SIZ_F algorithm
//Input: Priority queue  $Q[1, \dots, N_{\text{all}}]$ 
//Output: Status of source identification
if ( $N_{\text{all}} \geq N_{\text{ini}}$ )
  Sort  $Q$  in the order of the current up-flow direction
   $L[1, \dots, N_{\text{all}}] \leftarrow Q[1, \dots, N_{\text{all}}]$ ;  $n_1 \leftarrow N_{\text{all}}$  //  $L$  is a list
  status  $\leftarrow$  false
  while  $n_1 \geq N_{\text{min}}$  do
    Calculate  $(x_{\text{last}}^{(m)}, y_{\text{last}}^{(m)})$  of all LCDPs in the priority queue;
    Find  $p_{\text{max}}$  with  $D_{\text{max}}$  in Equation (5)
    if  $D_{\text{max}} > \varepsilon_F$ 
      remove  $p_{\text{max}}$  from  $L$ ;  $n_1 \leftarrow n_1 - 1$ 
    else
      status  $\leftarrow$  true; break
  if status = true
    return  $(x_{\text{last}}^{f(1)}, y_{\text{last}}^{f(1)})$  as the source location
  else
    return no source location identified
  else
    return no source location identified

```

maintain the plume characteristics that significantly complicate the plume tracing problems (intermittency, meander and varying flow) caused by natural flow fluid. Instead of adjusting the Reynolds numbers, it controls a filament release rate (5–10 filaments/s) to simulate filament intermittency and addresses the meandering nature of the plume as a key factor complicating the plume tracing. It also manipulates flow variation to challenge the CPT strategies. An OpArea is specified by $[0,100] \times [-50,50]$ in metres. The simulation time step is 0.01 s. A source location is chosen as (20, 0) in metres for checking the accuracy of identified source locations, but it is unknown to the tracer during CPT test runs. The simulation environment is set below: first, the filament release rate is 5 filaments/s because a low release rate may result in significant plume intermittency, which often causes the tracer to lose contact with the plume and consequently to make spurious identification. Second, the mean fluid velocity is 1 m/s. Figure 6(a) illustrates flow velocity varying from 0.86 to 1.17 m/s detected during a simulation run. The flow speed and variation, which are much larger than those detected during the in-water test runs, significantly challenge the efficacy and robustness of the plume tracing strategies. Next, measured fluid direction, which is corrupted by additive noise from a white normal random process, varies in $[-27.42, 28.30]$ in degree, as shown in the top panel of Figure 6(b). The bottom panel of Figure 6(b) displays an expanding scale of the flow direction for the time in $[184.3, 186.3]$ in second. These settings remain for all simulations in this study.

Figure 7 illustrates a plume-tracing trajectory of a test run from the first point of plume detection (marked by

a triangle) to the defined source location (marked by an asterisk) as well as odour-hit points. In order to present the difficulties of tracing the plume with significant meander in the larger scales of the OpArea, we address this test run in three regions of the OpArea: the significant meander region, the spurious identification region and the source identification region. The simulated plume exhibits significant meander, filament intermittency and low concentration, when it is transported over a significant distance (approximately 37 m in Figure 7) from its source. In this region, a tracer rarely directs toward the plume source despite its detection of an above threshold concentration. Successful plume tracing through this region assures tracking of the plume over distances of 100 m in the field tests. The concentration of the plume is still patchy, but plume meander is relatively smaller when the plume is advected to an intermediate distance (10–37 m in Figure 7) from its source. Most spurious identifications occur in this region because of significant filament intermittency, as shown in Figure 8. In the region within 10 m from the source, the tracer easily tracks the plume toward its source, but its manoeuvring in the vicinity of the source helps investigate the LCDPs characteristics for the source identification. For simulation test runs, the tracer velocity command is set as 1 m/s for Maintain-Plume. At this speed, which is close to the mean fluid velocity, we evaluate if Track-Out navigates the tracer to turn into a correct direction to encounter next chemical filament. The command for Reacquire-Plume is set 1.4 m/s, which is slightly greater than the mean fluid velocity to re-catch the lost plumes during Reacquire-Plume manoeuvres. The command 2 m/s for Find-Plume allows the tracer to explore the OpArea quickly. Figure 6(c) plots the

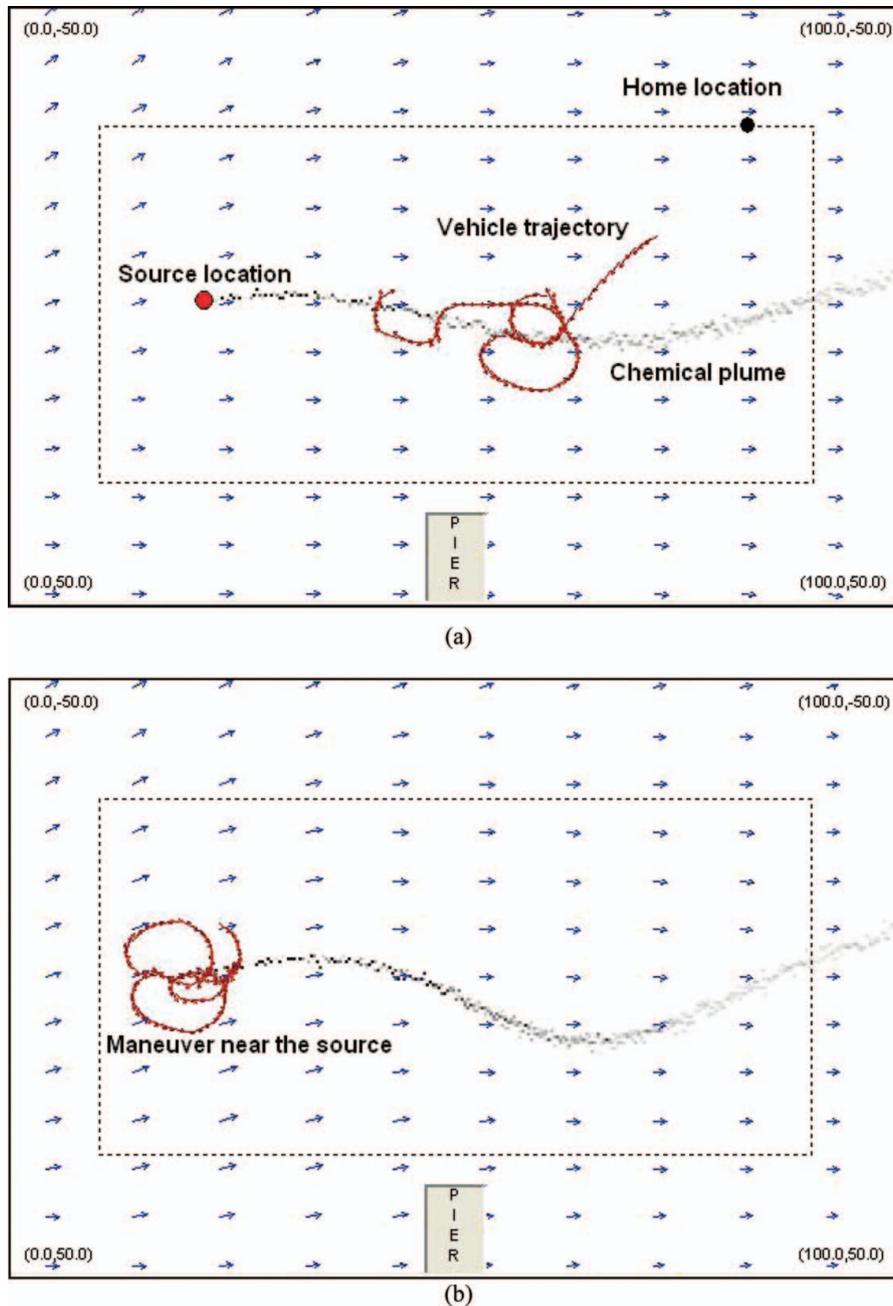


Figure 5. Olfactory-based chemical plume tracing in a simulated fluid-advected environment. The greyscale indicates above threshold concentration. The arrows indicate the magnitude and direction of the local flow vector at the tail of the arrow. (a) Tracer starts its CPT mission from the home location, including plume finding and plume tracing. (b) Tracer manoeuvres in the vicinity of the source location to identify the odor source.

tracer speed variations and transits between the different behaviours during a simulation test run.

4.2. Analysis of source identification algorithms

An identified source location is valid if its coordinates are situated within a given distance from a defined source

location; otherwise it is invalid (spurious). For the simulation evaluations, a valid source location is located within 10 m of the defined source location. Accordingly, identification time is the manoeuvring time prior to identifying the source location within the distance of the defined source location. Note that the SIZ algorithms do not interrupt Maintain-Plume and Reacquire-Plume activities

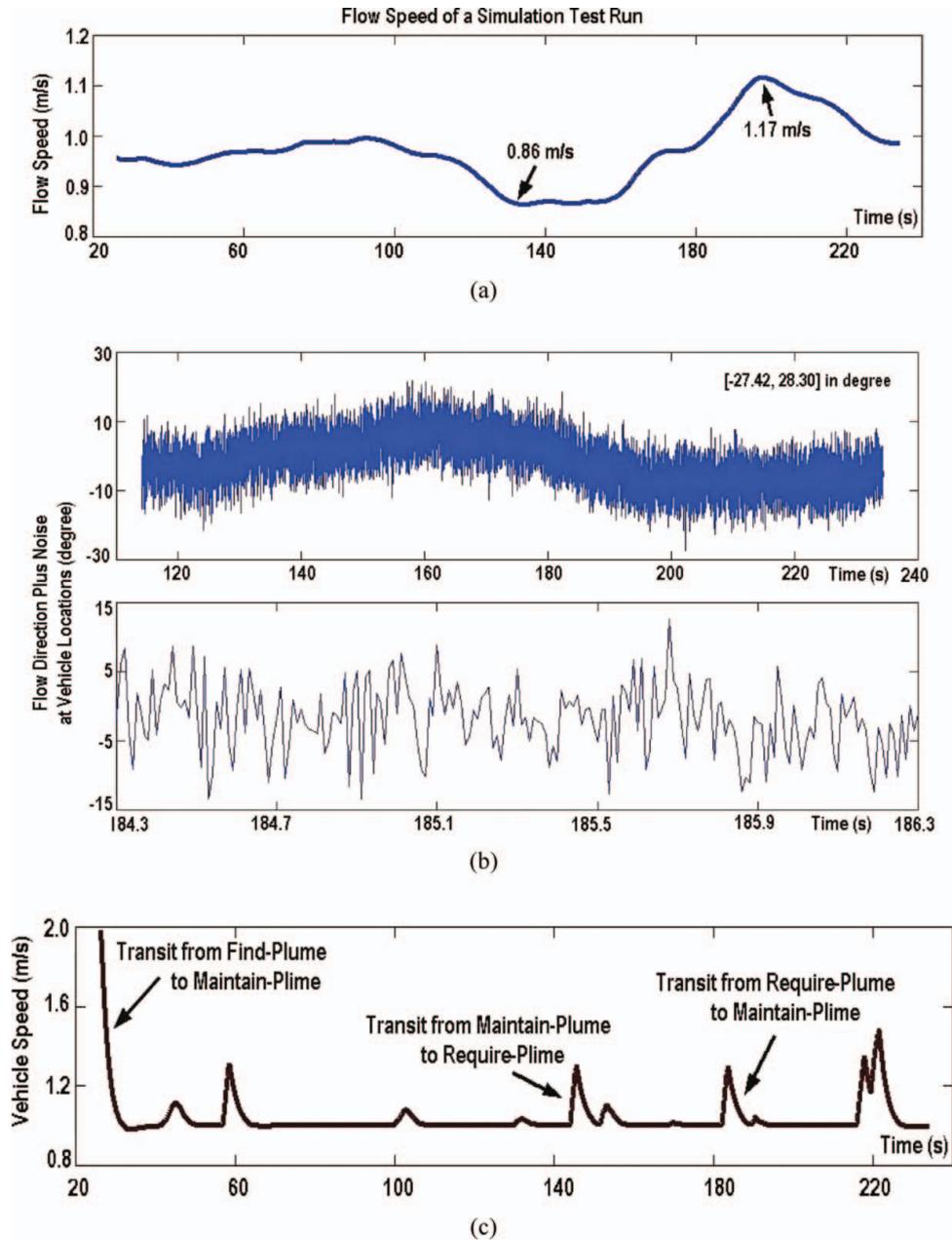


Figure 6. Flow information during a simulation test run. (a) Flow speed varies in $[0.86, 1.17]$ in m/s for t in $[26.69, 234.30]$ s during the test run. (b) Flow direction corrupted by additive white noise varies in $[-27.42, 28.30]$ in degree for t in $[26.69, 234.30]$ s during the test run (the top panel of the figure) and the expanding scale of the flow direction for t in $[184.3, 186.3]$ s. (c) Vehicle speed in m/s is documented for t in $[26.69, 234.30]$ s.

during CPT missions as they only use LCDPs to estimate the odour source. We propose a hypothesis: given an algorithm version, including its type and parameter setting, the mean identification time of valid source locations is almost independent of a tracer initial (a home location). To validate this hypothesis, we design two groups of evaluations with respect to initials $(80, -30)$ and $(40, 20)$ in metres. The two initials specify the different plume tracing characteristics.

Starting at $(80, -30)$ m, the tracer must trace the plume through the significant meander region while the traversal distance from $(40, 20)$ m to the source location is relatively shorter. On the basis of the Monte Carlo study results (the evaluation strategy will be explained in detail in section 5), we discuss the two algorithm versions with the parallel parameter values: SIZ_T with $\varepsilon_T = 6$ m and $N_{dec} = 6$, and SIZ_F with $\varepsilon_F = 6$ m and $N_{min} = 6$. These parameter

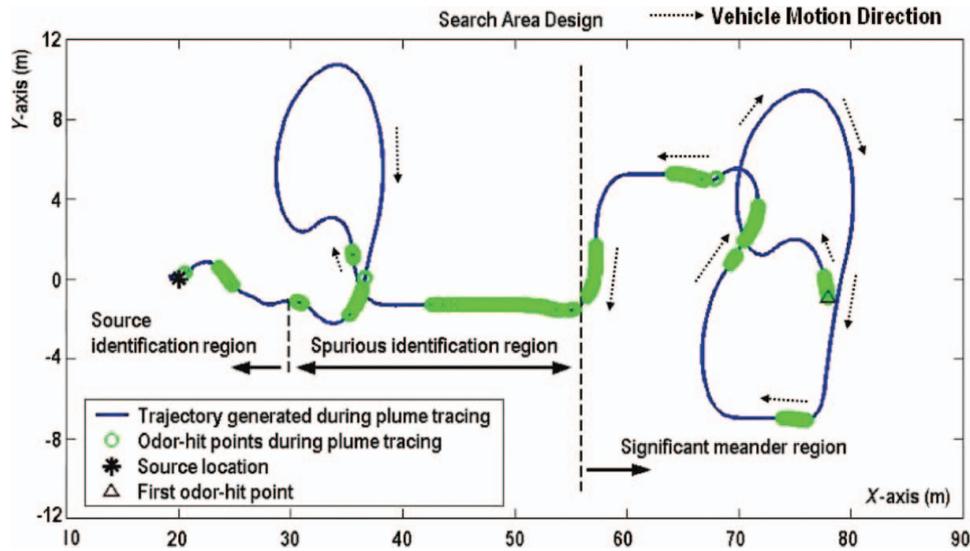


Figure 7. Tracer trajectory is documented from the first detection point to the identified source location during a simulation test run. The thick grey circles indicate locations where the chemical concentration was above threshold when the tracer was at that location. Tracer traces the plume toward its source through significant meander region, spurious identification region and source identification region.

settings achieve a high success rate of test runs because an intention of the validation is to evaluate the identification time of the valid source locations identified during CPT test runs. Because $N_{ini} = N_{dec}$ and $N_{ini} = N_{min}$, the initial value, $N_{ini} = 6$, is discarded. SIZ.T needs the last six most recent LCDPs close enough for the source identification, while SIZ.F needs to enclose the six most up flow LCDPs. For each group, the simulations continue 1000 CPT test runs without duplications of the trajectory, the odour-hit points or the LCDPs. We define a CPT test run as a cycle the tracer starts at its home location and returns the home location. The test run fails if the tracer cannot identify the source location within the time limit $T_{max} = 1000.0$ s (in simulation studies, this limit is used to measure CPT performance, while in the field tests a similar setting was defined to check remaining energy in order to bring the tracer back home); otherwise, it records the identification time, the total time for the test run and the coordinates of the identified source location. For each test run, both of the versions work concurrently, and the following situations may occur. First, both record an “over-time” test run, if they cannot identify the source location within T_{max} . Second, if one out of two first identifies the odour source within $T_{max} = 1000$ s, it is deactivated after reporting the identified source location, the identification time and the total time cost, but the tracer continues its CPT manoeuvre to identify the odour source using the next version. Finally, the tracer returns the home location, if the second version identifies the odour source within T_{max} again; otherwise, it returns to the home location with a record of an “over-time” test run.

Figure 9(a) shows that the SIZ.T version averages the total time of 388 s and 279.9 s marked by circles to

complete the CPT test runs with respect to (80, -30) m and (40, 20) m; while the SIZ.F version spends 359.5 s and 245.1 s. The two tracer initials apparently cause different times for plume tracing from the first chemical detection to the source. For the source identification, however, the SIZ.T version averages nearly the same time (218.5 s and 219.2 s marked by asterisks in Figure 9(a)) with respect to (80, -30) m and (40, 20) m. Similarly, the SIZ.F version averages nearly the same time (186.8 s and 183.9 s). The identification time of a valid source location relies only on the duration in which the tracer manoeuvres in the vicinity of the source location to detect sufficient LCDPs for source identification. The validated hypothesis provides a foundation for optimising the SIZ algorithms free from initial positions. The SIZ.T version yields standard deviations of 90.7 s and 95.3 s, while the SIZ.F version only yields standard deviations of 49.6 s and 46.1 s marked by squares in Figure 9(a).

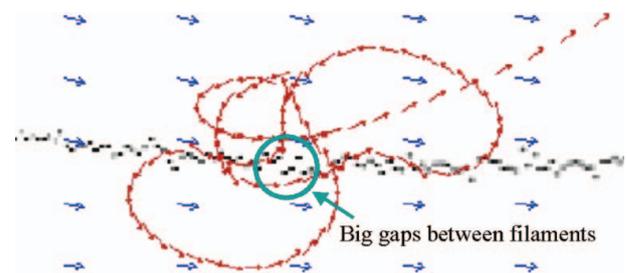


Figure 8. Tracking of the plume with significant filament intermittency may cause spurious source identification when the tracer clusters two LCDPs for source identification.

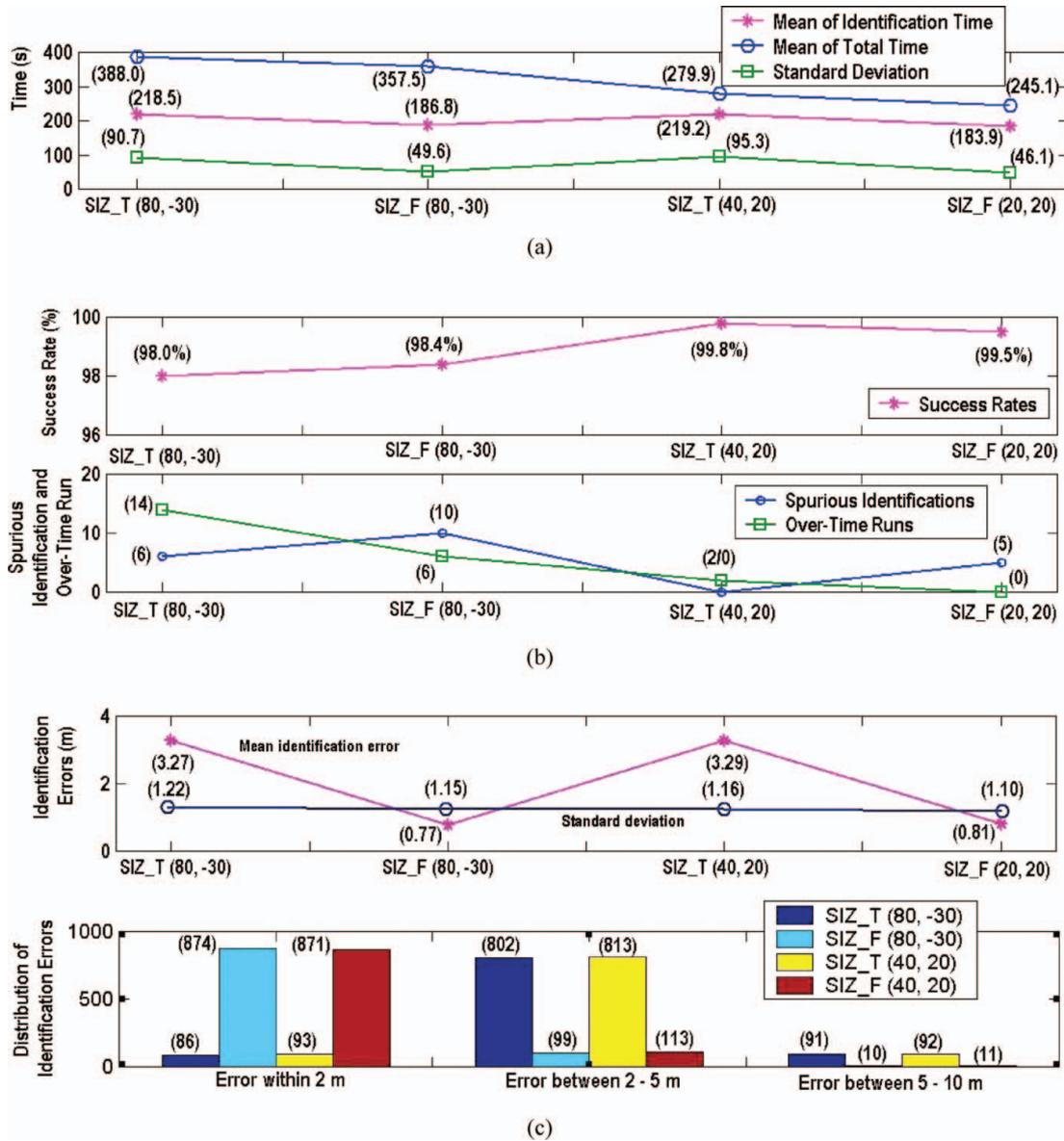


Figure 9. Evaluation results of SIZ.T and SIZ.F with respect to two initials (80, -30) and (40, 20) in metre for 1000 simulation test runs. (a) Mean total time, mean identification time and standard deviation of identification time for the 1000 test runs. (b) Success rate, spurious identifications and over-time runs of the 1000 test runs. (c) Mean identification error, standard deviation of identification errors and error distributions.

Figure 9(b) shows that both of the versions achieve success rates of over 98% with respect to the initial positions (40, 20) m and (80, -30) m. For (80, -30) m, SIZ.T makes six spurious identifications and fourteen “over-time” test runs during the 1000 CPT test runs, and SIZ.F makes ten spurious identifications and six “over-time” test runs. For (40, 20) m, SIZ.T makes no spurious identification and two “over-time” test runs. In contrast, SIZ.F makes five spurious identifications and no “over-time” test run.

The SIZ.F version records the mean identification error less than 1 m, while the SIZ.T version, about 3.27 m, as

shown in the top panel of Figure 9(c). The standard variation varies in [1.10, 1.22] in metre. The bottom panel of Figure 9(c) summarises a distribution of the identified source locations in three groups: within 2 m, between 2 and 5 m and between 5 and 10 m. The SIZ.F version achieves the high accuracy for the 1000 test runs. 88.82% of the valid source locations are located within 2 m, and only about 1% between 5 and 10 m. Using the SIZ.T version, however, only 8.78% of the identified source locations fall within 2 m and over 80.0%, between 2 m and 5 m. On the basis of the thorough analysis above, we choose the SIZ.F algorithm for optimisation.

5. SIZ.F Algorithm optimisation

The optimisation is designed to achieve a high success rate (reliability) with a rational identification time by investigating the SIZ.F parameters (ε_F , N_{\min} , N_{ini}). Note that the sole input to the SIZ algorithms is the priority queue maintaining LCDPs detected during a CPT test run. This feature leads to the development of a new strategy for optimisation. Here, we redefine a CPT test run by deactivating any source identification algorithm. The tracer starts at a home location, utilises Find-Plume, Maintain-Plume and Reacquire-Plume to move toward the source location and returns the home location *only when* the mission time reaches $T_{\max} = 1000$ s. We store LCDP (and the odour-hit points) detected during each test run into its associated log file. The 1000 CPT test runs; create the 1000 log files. For optimisation, we retrieve the LCDP nodes of each test run from its log file and insert them into the priority queue as the input to the SIZ.F algorithm. This strategy has two advantages: first, it evaluates different algorithm versions under identical experimental conditions, as it uses the same LCDPs of the CPT test runs; and second, it greatly reduces computational time for optimisation, as retrieving the LCDPs from the log files costs much less time than running 1000 CPT missions on-line. Figure 10 illustrates 38 LCDPs marked by red squares and 4066 odour-hit points marked by green circles detected during one test run. The LCDPs are only 9.35% of the odour-hit points, and most of them are clustered in the vicinity of the source location. On the basis of the retrieved LCDPs of the 1000 test runs, we conduct the three groups of evaluation.

Evaluations in Group 1 assess the effects of N_{ini} and N_{\min} on the identification performance. We fix $\varepsilon_F = 4$ m, vary N_{ini} from 4 to 5 and increase N_{\min} from 2 to 4. The evaluation results are interpreted with the following remarks. First, given $N_{\min} = 2$, changing N_{ini} from 4 to 5 results in increasing the success rate of 7.3% and the mean identification time of 5.5 s, indicated by two points (60.3, 51.9) and (67.6, 57.4) in the top panel of Figure 11(a). N_{ini} works as a filter to remove some invalid LCDPs. N_{\min} takes over N_{ini} , if N_{\min} is close to N_{ini} . For instance, given $N_{\min} = 4$, increasing N_{ini} from 4 to 5 makes only 0.1% increase of the success rate and no change in the identification time, as indicated by two points (95.1, 157.0) and (95.2, 157.0) in the top panel of Figure 11(a). $N_{\text{ini}} = 4$ is fixed for the subsequent evaluations. Second, increasing N_{\min} significantly improves the success rate. The two curves of Group 1 in Figure 11(a) show that, given $N_{\text{ini}} = 4$ or $N_{\text{ini}} = 5$, changing N_{\min} from 2 to 4 results in increasing the success rate from 60.3% through 86.3% to 95.1% or from 67.6% through 88.9% to 95.2%, as increasing N_{\min} significantly decreases the spurious identifications from 396 through 136 to 44 or from 321 through 108 to 42 in the top panel of Figure 11(b). Third, increasing N_{\min} by one results in increasing the identification time about 50 s, denoted by a time constant, κ ,

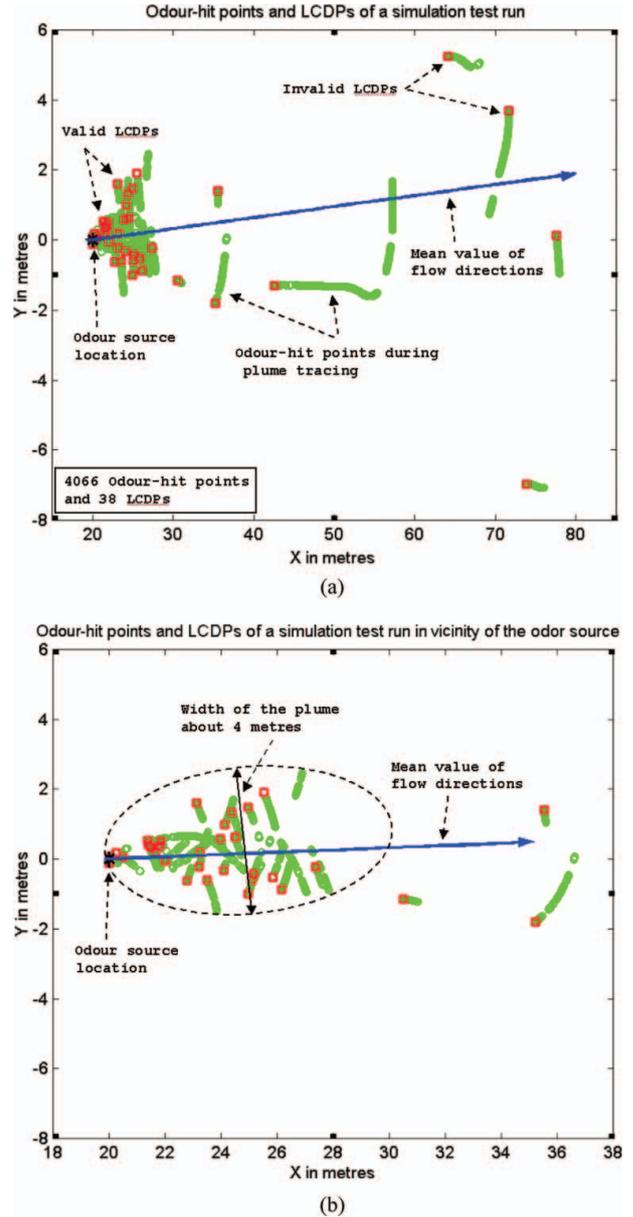
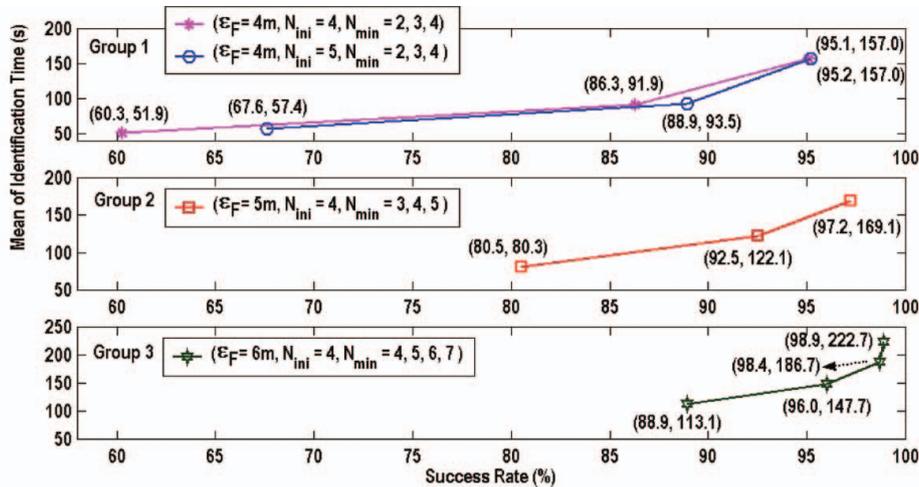
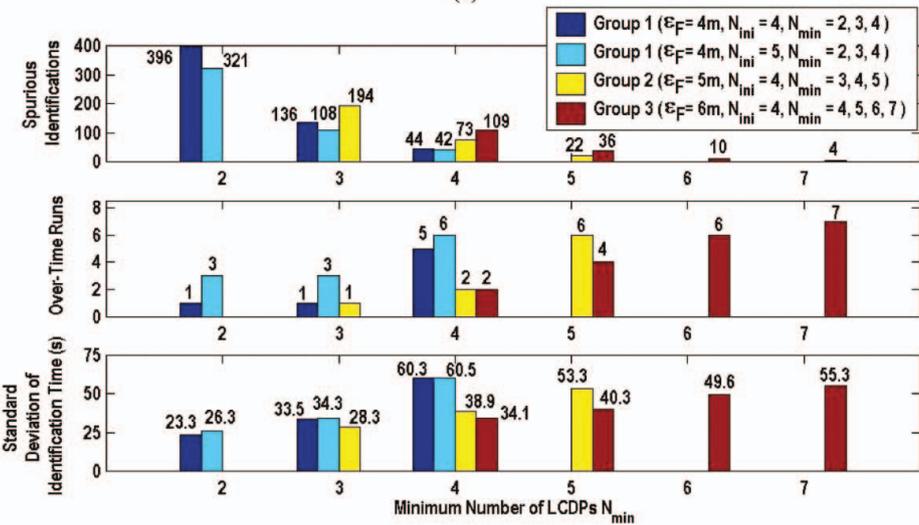


Figure 10. Distribution of odour-hit points and LCDPs of a simulation test run. (a) 4066 odour-hit points and 38 LCDPs detected during the test run. (b) Expanding scale of odour-hit points and LCDPs in the vicinity of the odour source.

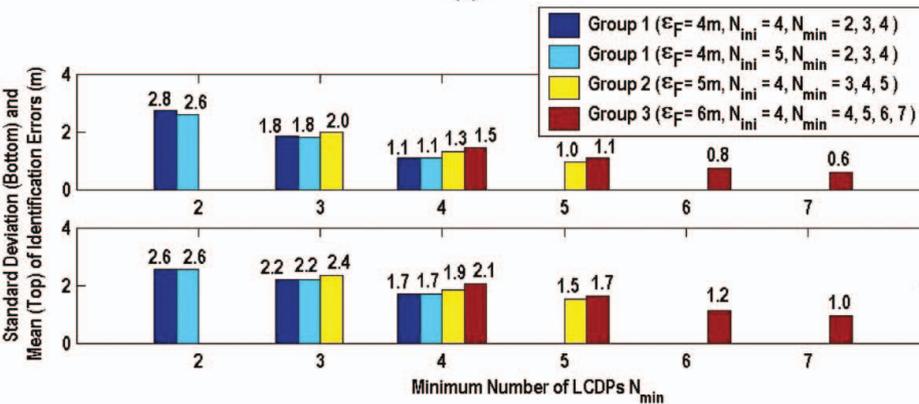
which allows the tracer to traverse at least one cloverleaf with a diameter of 10–15 m to detect next LCDP during a Reacquire-Plume activity. We propose a time metric $(N_{\min} - 1) \times \kappa$ to estimate the identification time. Note that $(N_{\min} - 1) \times \kappa$ provides further evidence of identification time free from tracer initials. For example, when $N_{\min} = 4$, the estimated time of 150 s is close to the evaluation result of 157 s. In Group 1, $N_{\min} = 2$ achieves the low identification time of 51.9 s and 57.4 s, close to the estimated value of 50 s, but the success rate of about 60% is not acceptable. Therefore, $N_{\min} = 2$ is excluded in the following evaluations.



(a)



(b)



(c)

Figure 11. Evaluation results of SIZ_F optimisation for 1000 simulation test runs. (a) Mean identification time versus success rate for the 1000 test runs. (b) Spurious identifications, over-time runs and standard deviation of identification time for the 1000 test runs. (c) Mean identification error and standard deviation of identification error.

Three evaluations in Group 2 further detect the effects of N_{\min} on the identification performance at $\varepsilon_F = 5$ m. This group takes $N_{\text{ini}} = 4$, as recommended above, and increases N_{\min} from 3 to 5. The evaluation results show that the N_{\min} increases results in improving the success rate from 80.5% through 92.5% to 97.2% at a given ε_F , but the mean identification time increases from 80.3 s through 122.1 s to 169.1 s in the curve of Group 2 in Figure 11(a). However, increasing N_{\min} from 4 to 5 makes an improvement of the success rate only 4.7%. Comparing Group 2 with Group 1 for $N_{\min} = 3$ or $N_{\min} = 4$, increasing ε_F from 4 m to 5 m results in reducing the mean identification time from 91.9 s to 80.3s or from 156.99 s to 122.09 s but results in decreasing the success rate from 86.3% to 80.5% or from 95.1% to 92.5%.

Four evaluations in Group 3 further investigate the effects of N_{\min} on the source identification performance at $\varepsilon_F = 6$ m. This group also takes $N_{\text{ini}} = 4$, and changes N_{\min} from 4 to 7. Represented by the curve of Group 3 in Figure 11(a), the evaluation results show that increasing N_{\min} from 4 to 7 results in improving the success rate from 88.9% through 96.0% and 98.4% to 98.9% but increasing the mean identification time from 113.1 s through 147.7 s and 186.7 s to 222.7 s. Note that increasing N_{\min} from 4 to 5 or from 5 to 6 improves only the success rate of 2.4% or 0.5%. This result indicates that the success rate nearly reaches its limit when $N_{\min} > 6$. Comparing Group 3 with Group 2 for a given N_{\min} , increasing ε_F from 5 m to 6 m results in reducing the mean identification time but decreasing the success rate.

We summarise the evaluations in Groups 1–3. First, a high success rate goes along with a high identification time, as shown in Figure 11(a). In application, achieving a high success rate should have priority. Here, we recommend $N_{\min} = 4$ or $N_{\min} = 5$, since a larger N_{\min} cannot significantly improve the success rate and unnecessarily increases the identification time. Increasing N_{\min} causes a slight increase of “over-time” runs shown in the middle panel of Figure 11(b). Second, a bigger ε_F makes the source identification quicker but causes a lower success rate. Figure 10(b) illustrates the expanded scale of the vicinity of the source location. The estimated plume width over 4 m suggests $\varepsilon_F = 5$ m or $\varepsilon_F = 6$ m. These settings from Groups 2 and 3 result in the standard deviation of the identification time in [34.1, 55.3] s, as shown in the bottom panel of Figure 11(b). Next, the mean and standard deviation of identification errors in a scope from 2.77 m to 0.59 m are satisfied, as shown in Figure 11(c). Finally, N_{ini} is ignored because $N_{\text{ini}} = 4 \leq N_{\min}$. The optimised algorithm achieves the mean identification time in about 3 min and a success rate of about 90%.

6. Discussion and conclusion

Insect-inspired navigational strategies were reviewed by Cardé and Willis (2008). We abstract the two SIZ source identification algorithms from the moth-inspired plume tracing strategies reported by Li et al. (2001) and use Monte

Carlo methods to evaluate and optimise the source identification algorithms in a simulated plume environment (Farrell et al. 2002). In comparison with plume tracing experiments in a laboratory environment, simulation studies provide the following advantages: (1) Using the simulated plume, different CPT strategies can be evaluated under the uniform conditions. (2) Tracing the simulated plume is much more difficulty than laboratory plumes, since the fluid flow directions and magnitudes in the simulated environment vary in location and time, while wind directions and magnitudes in the laboratory environment remain nearly unchanged (Kowadlo and Russell 2003). In fact, odour source in a laboratory environment can be easily localised with point-by-point searching strategies. (3) Flow speed and flow variations in the simulation runs, which are much larger than those detected during the in-water test runs (Li 2007), disperse the chemicals rapidly to challenge the efficacy and robustness of plume tracing strategies.

The SIZ_T condition for source declaration is stronger than the SIZ_F condition. SIZ_T needs the six most recent LCDPs close enough for source declaration, while SIZ_F needs the six most up flow LCDPs close enough. The SIZ_T condition requires a robot to overshoot nearly the same position in the vicinity of the odour source for the last six times, but a number of factors can disperse LCDPs, such as the width of chemical plumes, the variation of flow directions, the intermittency of filaments or the robot’s mechanical restrains. However, by using the SIZ_F algorithm, it is relatively easier to enclose the six most up flow LCDPs inside SIZ_F, since the odour source located in the up flow direction always makes the robot an up flow progress. The result in Figure 9(a) shows that SIZ_F, in average, saves about 14.5% energy to complete source declarations.

The simulation studies show that a success rate in declaring source locations reaches about 90%, the average time cost of the identified source locations is about 3–4 min and the average error is about 1–2 m for 1000 test runs in an operation area with length scales of 100 m. This success rate is much higher than the 70% rate from tracing a pheromone plume in insects (Belanger and Arbas 1998). The CPT strategies investigated in this paper have potential for extension to trace plumes in three dimensions. Our further research will address simulation evaluations of tracing a chemical plume in 3-D.

Acknowledgment

Jay Farrell, Ring Cardé and John Murlis provided valuable insight into the processes of moth-inspired plume tracing strategies. The author thanks his students with the Department of Computer Science at California State University, Bakersfield, for their efforts in performing simulation test runs.

References

- Belanger JH, Arbas EA. 1998. Behavioral strategies underlying pheromone-modulated flight in moths: lessons from simulation studies. *J Comp Phys.* 183:345–260.

- Cardé RT. 1996. Odour plumes and odour-mediated flight in insects in olfaction in mosquito-host interactions. Proceedings of CIBA Foundation Symposium. 54–66. John Wiley & Sons Ltd, Netherlands.
- Cardé RT, Willis MA. 2008. Navigational strategies used by insects to find distant, wind-borne sources of odor. *J Chem Ecol.* 34:854–866.
- Cowen EA, Ward KB. 2002. Chemical plume tracing. *Environ Fluid Mech.* 2(1–2):1–7.
- Farrell JA, Murlis J, Long X, Li W, Cardé RT. 2002. Filament-based atmospheric dispersion model to achieve short time-scale structure of chemical plumes. *Environ Fluid Mech.* 2(1–2):143–169.
- Farrell JA, Pang S, Li W. 2005. Chemical plume tracing via an autonomous underwater vehicle. *IEEE J Ocean Eng.* 30:428–442.
- Grasso FW, Atema J. 2002. Integration of flow and chemical sensing for guidance of autonomous marine robots in turbulent flows. *Environ Fluid Mech.* 1:1–20.
- Grasso FW, Consi TR, Mountain DC, Atema J. 2000. Biomimetic robot lobster performs chemo-orientation in turbulence using a pair of spatially separated sensors: progress and challenges. *Rob Auton Syst.* 30:115–131.
- Hayes AT, Martinoli A, Goodman RM. 2002. Distributed chemical source localization. *IEEE Sens J.* 2:260–271.
- Ishida H, Kagawa Y, Nakamoto T, Moriizumi T. 1996. Chemical-source localization in the clean room by an autonomous mobile sensing system. *Sens Actuators B.* 33:115–121.
- Ishida H, Nakamoto T, Moriizumi T, Kikas T, Janata J. 2001. Plume-tracking robots: a new application of chemical sensors. *Biol Bull.* 200:222–226.
- Kowadlo G, Russell RA. 2003. Naïve Physics for Effective Odour Localisation. Proceedings of Australian Conference on Robotics and Automation.
- Li F. 2009. Multi-Robot Odor-Source Localization in Turbulence Dominated Airflow Environments [Ph. D. Dissertation]. [Tianjin (China)]: School of Electrical Engineering and Automation, Tianjin University.
- Li W, Farrell JA, Cardé RT. 2001. Tracking of fluid-advected chemical plumes: strategies inspired by insect orientation to pheromone. *Adapt Behav.* 9:143–170.
- Li W, Farrell JA, Pang S, and Arrieta RM. 2006. Moth-inspired chemical plume tracing on an autonomous underwater vehicle. *IEEE Trans Robot.* 22(2):292–307.
- Li W. 2006. Abstraction of odor source identification algorithm from moth-inspired plume tracing strategies. Proceedings of IEEE Conference on Robotics and Biomimetics. 1024–1028.
- Li W. 2007. Moth Plume-Tracing Derived Algorithm for Identifying Chemical Source in Near-Shore Ocean Environments. Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, USA. pp. 2162–2167.
- Liao Q, Cowen EA. 2002. The information content of a scalar plume – A plume tracing perspective. *Environ Fluid Mech.* 2(1–2):9–34.
- Lilienthal AJ, Ulmer H, Froehlich H, Werner F, Zell A. 2004. Learning to Detect Proximity to a Gas with Mobile Robot. Proceedings of IEEE Conference on Intelligent Robots and Systems, Sendai, Japan. 1444–1449.
- Marques L, Nunes U, de Almeida AT. 2002. Olfaction-based mobile robot navigation. *Thin Solid Films.* 418:51–58.
- Murlis J. 1986. The structure of odour plumes. In: Payne TL, Birch MC, Kennedy CEJ, editors. Mechanisms in insect olfaction, Oxford University Press, Oxford, UK. p. 27–38.
- Russell RA. 2001. Tracking chemical plumes in constrained environments. *Robotica.* 19:451–458.
- Webster DR, Rahman S, Dasi LP. 2001. On the usefulness of bilateral comparison of tracking turbulent chemical odor plumes. *Limnol Oceanogr.* 46:1048–1053.
- Weissburg MJ, Dusenbery DB, Ishida H, Janata J, Keller T, Roberts PJW, Webster DR. 2002. A multidisciplinary study of spatial and temporal scales containing information in turbulent chemical plume tracking. *Environ Fluid Mech.* 2(1–2):65–94.