BRINGING ARTIFICIAL OLFACTION AND MOBILE ROBOTICS CLOSER TOGETHER – AN INTEGRATED 3D GAS DISPERSION SIMULATOR IN ROS

*Ali Khaliq, Sepideh Pashami, Erik Schaffernicht, Achim Lilienthal and Victor Hernandez Bennetts**

AASS – School of Science and Technology – Örebro University

*****victor.hernandez@oru.se

ABSTRACT

Despite recent achievements, the potential of gas-sensitive mobile robots cannot be realized due to the lack of research on fundamental questions. A key limitation is the difficulty to carry out evaluations against ground truth. To test and compare approaches for gas-sensitive robots a truthful gas dispersion simulator is needed. In this paper we present a unified framework to simulate gas dispersion and to evaluate mobile robotics and gas sensing algorithms using ROS. Gas dispersion is modeled as a set of particles affected by diffusion, turbulence, advection and gravity. Wind information is integrated as time snapshots computed with any fluid dynamics computation tool. In addition, response models for devices such as Metal Oxide (MOX) sensors can be integrated in the framework.

Index terms– Mobile robot olfaction, gas dispersion simulation, gas sensor simulation, MOX sensors, environmental monitoring.

1. INTRODUCTION

Gas sensing is critical in different applications. For example, biogas producers must report their emission levels due the high global warming impact of CH_4 [1]. With considerable amounts of unaccounted CH_4 emissions, as reported by the Environmental Protection Agency of the United States of America[1], the development of effective sensing mechanisms is of paramount importance. In this context, gas sensitive mobile robots can bring considerable advantages during inspection routines.

Mobile Robot Olfaction (MRO) studies the integration of gas sensing capabilities on board mobile platforms. An MRO system can perform automated inspection routines with a high spatial resolution. The flexibility of mobile robots allow for adaptable data collection that in turn can be used to quantify emission levels and to predict the location of emitting gas sources. Moreover, mobile robots can be rapidly deployed in emergency situations where hazardous gases, such as H_2S, might be present.

Validation of MRO systems is complex and time consuming. Preparations have to be made at the start and at the end of each trial. This results in a reduced number of trials that do not allow for statistical validation. Repeatability is also an issue since gas dispersion is a complex phenomenon that depends on several factors such as ambient temperature, wind and topographic conditions [2]. Slight changes in these environmental variables are expected and thus, the outcome between experiments can substantially diverge when e.g. the wind pattern changes.

It is thus of high importance to have a reliable simulator to evaluate the algorithms, sensing mechanisms and the robotic platform itself. Once the MRO system has been substantially evaluated under computer simulations, experimental trials can then be carried out in real world scenarios.

This paper presents a modular simulation framework aimed at the development of MRO systems. The presented software allows the user to model critical aspects such as the robotic platform, the gas sensing mechanisms, the wind flow patterns, the configuration of physical obstacles, the gas dispersion process and the properties of the released chemicals. Of particular interest for roboticists is that the presented simulator was implemented entirely using the Robot Operating System (ROS). ROS is an open source initiative that aims to implement a flexible framework for writing robot software and it is arguably the most widely used operating system used in robotics research. The flexibility of ROS allows the presented framework to simulate aspects beyond gas sensing such as path planning, robot manipulation, sensing mechanisms (e.g. range sensing, odometry) and the physical properties of the robots themselves[2].

The rest of this paper is structured as follows: Section 2 introduces related work, Section 3 presents the software implementation. In Section 4, simulation runs are shown while Section 5 presents the conclusions of this work.

2. RELATED WORK

Filament-based [3,4] and *puff-based* [5] simulators have been proposed to create macro-scale models that predict the dispersion of e.g. radioactive particles. While these algorithms can produce sophisticated models, they do not allow the possibility to simultaneously validate robotic platforms and gas sensing mechanisms. An example of a dispersion simulator for robotics applications is *Plumesim* [6], originally developed for the now outdated player/stage robotics framework. *Plumesim* implements simplistic pre-defined dispersion models and in order to generate more complex models, that can include e.g. obstacles, it requires the use of licensed fluid dynamics software such as ANSYS[3]. More recently, Gonzalez et al implemented a gas dispersion simulator for MRO [7] using the OpenMORA robotic OS and gas dispersion snapshots, computed with an external tool, as an input. However, a drawback of the simulator presented in [6] is that OpenMORA[4] lacks of the support provided by more widespread frameworks like ROS.

3. IMPLEMENTATION

Figure **1.A** shows a block diagram of the implemented simulator. The light blue boxes correspond to the simulator inputs that are used by the different internal modules. The robot and environment models contain a set of descriptors that specify the sensing and actuating mechanism of the mobile platform and the spatial configuration of the environment. The launch file contains a series of parameters, for example, the location of the emitting gas source, the size of the exploration area and the type of chemical being released.

[1] http://spectrum.ieee.org/energy/fossil-fuels/satellites-and-simulations-track-missing-methane

[2] http://www.ros.org/

[3] http://www.ansys.com/

[4] https://github.com/OpenMORA

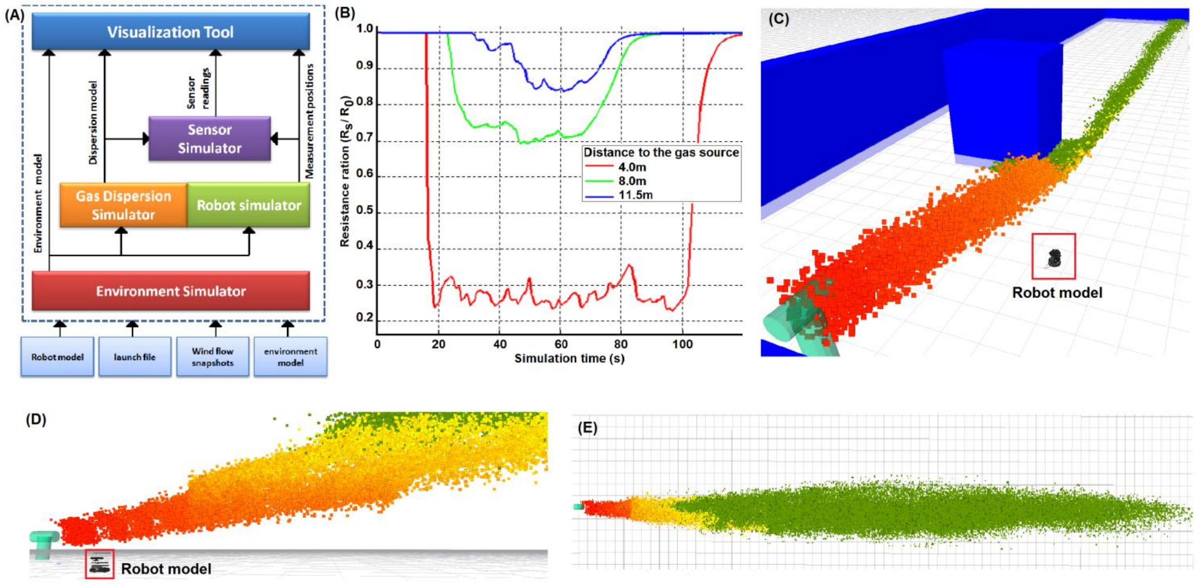

Figure 1: (A) Block diagram of the simulator (B) Simulated responses of a TGS2611 MOX sensor. (C) Simulation run with an ethanol gas source and physical obstacles. (D) and (E), 3D and top-down views of a CH_4 simulation run. Red shades denote high concentrations while green denote low concentration levels. In both environments, a toy-like robot is simulated. The cyan markers denote the position of the gas sources.

A 3D extension based on the work of Pashami and co-authors [8] is presented in this work. In [8], particles are subject molecular diffusion, turbulence and advection. Turbulence is modeled as snapshots of the wind flow at given time intervals. In the presented 3D extension, gravity is considered to compute the particle's acceleration. The sensor simulation block models the sensing mechanisms (e.g. remote or in-situ gas sensors). In this work, three commercially available gas sensor models were implemented using real world data to fit exponential response curves. The visualization block corresponds to the native ROS node "RVIZ", that allows to visualize data, spatial configurations and robot models in a user friendly interface.

4. RESULTS

Figures **1.C**, **1.D** and **1.E** show simulation runs using the implemented framework. The turbulent wind data was generated with OpenFOAM[5]. However, the selection of the fluid dynamics tool is implementation free. In Figures **1.D** and **1.E**, a source of CH_4 was placed in an obstacle free area. The density of CH_4 allows for a buoyant plume. In Figure **1.C**, an ethanol source is inside an area with obstacles. Ethanol is denser than air and thus propagates at ground level. Notice the meandering of the plume near the location of the obstacle. Figure **1.B** shows the simulated exponential response model of a set of TGS2611 MOX sensor placed at 4m, 8m and 11.5m from a CH_4 source. The delay within the sensor responses is correlated with the distance to the gas source. The ratio R_s/R_0, (R_s being the instantaneous resistance and R_0 the baseline), decreases according to the concentration. The sensor show faster response times compared to the recovery times. This is due to the different time constants in the rise and decay, which is common for MOX sensors.

5. CONCLUSIONS

We presented a 3D gas dispersion simulator aimed to Mobile Robot Olfaction. The main contribution of this work is the possibility of integrating both, robotics and gas dispersion simulation in a single open source ROS module[6]. ROS offers a wide variety of sensing modalities, robotic algorithms and platform models that can be incorporated to simulate MRO systems. In this way, gas sensitive robotic systems can be extensively validated with simulations before carrying out time consuming real world experiments. Future work includes the use of the proposed framework in the evaluation of leak detection algorithms with remote and in-situ sensors.

6. REFERENCES

[1] Scharff, H. Achieving Adequate Control of Landfill Gas in Europe. Waste Management World, 2008.

[2] B.I Shraiman, E.D. Siggia, Scalar Turbulence, Nature (2000), Vol 405, pp. 639–646.

[3] Graff, A. A Lagrangian particle dispersion model. 8th Int. Conf. on Harmonisation within Atmosperic Dispersion Modelling for Regulatory Purposes, Bulgaria, 14–17. 2002.

[4] J. Farrell, J. Murlis, X Long, W Li, R. Cardé, Filament-based atmospheric dispersion model to achieve short time-scale structure of odor plumes. Environmental Fluid Mechanics. Vol 2. Pp. 143-169. 2002

[5] Thykier-Nielsen, S.; Deme, S.; Mikkelsen, T. Description of the Atmospheric Dispersion Module RIMPUFF; Technical Report. Risø National Laboratory. Denmark, 1999.

[6] G. Cabrita, P. Sousa and L. Marques. PlumeSim - Player/Stage Plume Simulator. ICRA 2010 Workshop for Networked and Mobile Robot Olfaction in Natural, Dynamic Environments, Anchorage, Alaska, USA, 2010

[7] J. G. Monroy, J.L. Blanco, J. Gonzalez-Jimenez, "An Open Source Framework for Simulating Mobile Robotics Olfaction", 15th ISOEN, Daegu, South Korea, 2013.

[8] S. Pashami, S. Asadi and A. Lilienthal. Integration of OpenFOAM Flow Simulation and Filament-Based Gas Propagation Models for Gas Dispersion Simulation. 4th Op. source CFD International Conference. Germany, 2010.

[5] http://www.openfoam.com/

[6] The repository for the implemented gas dispersion simulation framework is available under the following link: http://bitbucket.org/vhbennetts/gas_dispersion_simulator.git