Multiagents and Pollutant Monitoring

João Costa Seco*[1], Carlos Pinto-Ferreira[2], and Luís Correia[1]

[1] Departamento de Informática
Faculdade de Ciências e Tecnologia
Universidade Nova de Lisboa
2825 Monte da Caparica, Portugal
phone: 315-1-2948596
fax: 315-1-2948541
{jcs,lc}@di.fct.unl.pt

[2] Instituto de Sistemas e Robótica
Instituto Superior Técnico
Av. Rovisco Pais, 1
1096 Lisboa Codex, Portugal
phone: 315-1-8418270
fax: 315-1-8418291
cpf@isr.ist.utl.pt

Topic areas: Intelligent information retrieval, Multi-agent and distributed problem-solving, Robotics.

Abstract. The purpose of controlling pollutant emissions is important in environmental quality preservation. However, in order to control, a previous step including pollutant measurement and assessment should be performed.
What this paper suggests is the application of a robotic multiagent system to evaluate pollutant levels in the neighbourhood of one or more pollutant sources. In generic terms, this is the problem of searching and sampling a set of variables within a large dimension environment. The implemented case study is the simulation of a cloud produced by a power-plant chimney monitored by a set of helicopter models equipped with sensor and processing devices.
These robotic agents are controlled by a behavioural architecture and communicate through a simple signalling process. They have two kinds of behaviours: *individual* ones, which permit them to accomplish the mission on their own and *group* behaviours that increase the society performance by gathering and spreading agents around interesting locations. Both sensors and behaviours are quite elementary, however, they are able to produce good results, showing that such a complex problem may be solved with a low congnition approach.
The paper describes some simulation results on the reconstruction of pollutant clouds generated according to different environment conditions.

Other keywords: Agents, Agent societies, Autonomous robots, Simulation models, Pollutant monitoring

* Supported by JNICT, scholarship no. BM2902 from "Programa de mobilidade de recursos humanos".

Multiagents and Pollutant Monitoring

Abstract. The purpose of controlling pollutant emissions is important in environmental quality preservation. However, in order to control, a previous step including pollutant measurement and assessment should be performed.
What this paper suggests is the application of a robotic multiagent system to evaluate pollutant levels in the neighbourhood of one or more pollutant sources. In generic terms, this is the problem of searching and sampling a set of variables within a large dimension environment. The implemented case study is the simulation of a cloud produced by a power-plant chimney monitored by a set of helicopter models equipped with sensor and processing devices.
These robotic agents are controlled by a behavioural architecture and communicate through a simple signalling process. They have two kinds of behaviours: *individual* ones, which permit them to accomplish the mission on their own and *group* behaviours that increase the society performance by gathering and spreading agents around interesting locations. Both sensors and behaviours are quite elementary, however, they are able to produce good results, showing that such a complex problem may be solved with a low congnition approach.
The paper describes some simulation results on the reconstruction of pollutant clouds generated according to different environment conditions.

Topic areas: Intelligent information retrieval, Multi-agent and distributed problem-solving, Robotics.

1 Introduction

Multiagent systems operating on real environments is a challenging application in the field of agent theory. The major issue of this work is to supply tools for real implementations by the means of a simulated reactive multiagent system.

These agents have a simple mission that is to monitor a pollutant cloud originated from a power plant chimney by sampling the current pollutant values. A global map is built on a central processing unit on the ground. This map is then transformed into an image that holds information about cloud direction, pollutant concentration, and so on. This allows decision makers to evaluate and change the burning conditions of the power plant.

Present cloud monitoring approaches utilise a set of land sensor stations that transmit their readings to a central processing unit and by some means of estimation, values are predicted and decisions are made. The use of robotic agents to search and sample a polluted area (on the air, on the ground or underwater) can be a way that permits direct assessment of the real values and a more accurate monitoring process.

However, the sampling of a large area is not an easy job. A systematic or random search process may either take a while or require a large number of agents. On the other hand, the use of "intelligent" agents reduces the number of agents needed to perform such a task in a reasonable period of time. An interpolation process is viable and gives more interesting results if samples are taken in "interesting" locations.

This so called "intelligence" can be obtained by a reactive architecture based on behavioural modelling with dynamic decision making [6]. This kind of architecture is designed to deal with multiple sensors, multiple goals being robust in non-structured environments [3, 4].

To assess the efficiency of the social behaviour, a simulator creates a modelled pollutant cloud, "launches" a set of robotic agents that sample the cloud using several group behaviour strategies and the collected clouds are compared to the one that was generated and monitored. This allows a decision on what is the best behaviour configuration for such a group of agents.

Group strategies were inspired on Kube and Zhang's work on minimal group behaviours. These concern a non-interference strategy as a basis for more complex behaviours such as follow or gather [11].

The use of simulation gives the chance of testing this kind of behaviours without building the real agents. However, this is only the iceberg's tip, the gap between simulation and reality represents a large amount of work. So, options on computational models were taken in order to keep it at a simple level as a way to concentrate attention on behaviours design.

2 Environment Description

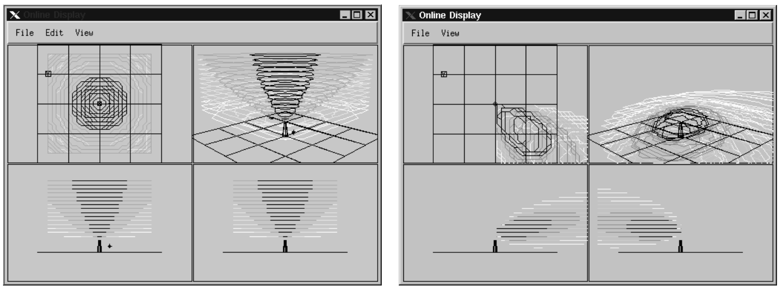
The whole system is composed of three parts: The world simulator, the agents, and the processing unit. The first of all is in charge of all computational models, from helicopters to pollutant and it gives agents and the processing unit a three dimensional environment to work on. Agents take actions based on sensor readings and the processing unit generates clouds based on samples taken by the agents.

2.1 Pollutant Modelling

There are two aspects on pollutant modelling in this simulation: producing a cloud to be monitored by the agents and reconstructing a cloud from the set of samples taken by them. A third aspect is the cloud visualisation.

Producing Pollutant Clouds An analytical formula from the laminar jet dispersion theory [17] was used in order to simulate a cloud produced by a power plant chimney. This creates a cloud like the one represented on Figure 1(a). The cloud has a paraboloid shape and surrounds completely the chimney area. This means that the higher value is always located at the chimney exit and along the chimney axis.

One the other hand, a more sofisticated and harder to monitor model was experimented in order to verify the agents robustness. This is a statistical gaussian dispersion model and it is based on a wind vector and a dispersion class. The pollutant values decrease uniformly along the wind vector as presented in Figure 1(b) [8]. On the simulator



(a) Laminar Jet Cloud

(b) Gaussian Cloud

Fig. 1. Pollutant Generator Models

structure there is a separate module that feeds the pollutant reader sensor according to the agent locations and chosen pollutant model[1].

Pollutant Sampling and Cloud Building Algorithm All samples taken by the agents are gathered with the information of the pollutant value, sampling time and sample location. They are then used to calculate a three dimensional grid that contains all the samples taken and has a fixed step[2]. The interpolation used to obtain the value grid is a quadratic one and it is applied after a selection of some eligible points. The eligible point criterion is dictated by a maximum number of points ordered by proximity within a defined range[3].

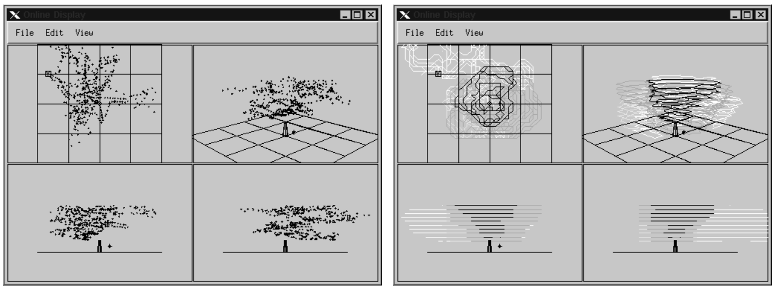
This grid is then passed through a discretisation procedure into several levels of pollutant value. The visualisation is obtained by a contour extraction algorithm (chain-coding) applied on each horizontal layer of the discrete grid. On the original cloud situation the grid is filled directly from the pollutant model.

2.2 Central Processing Unit

This component is a ground facility that receives all the readings from the agents and produces the expected global result. So, in this context there is no individual result for each agent, there are only social results. This set of samples is processed in order to produce useful information for the power plant decision makers.

Along with the sample values, the processing unit also registers the agents global location for each sample and the calculated cloud. This location can be obtained by a triangulation method and the global cloud is computed by gathering the samples from all agents and performing data interpolation.

[1] The pollutant model to be used is a run-time parameter for the simulator.
[2] The value used in this simulation is about 50m.
[3] The six closest points within a range of 200m are considered to be eligible for every point on the grid.



(a) Collected samples (b) Interpolated cloud

Fig. 2. Agent Society Results

2.3 Agents

Agents are small helicopter scale models equipped with perception, actuation and computational devices. Based on this equipment, there is a high level control architecture that must be capable of performing a mission. This control is built according to an architecture which is a way to implement an individual and social behaviour adequate to the mission goals.

There are options taken in all these topics and this description intends to present the agent's platform. Perception and actuation are important parts of the agent because all other options depend on them. Therefore, they will be the first to be described. Other important part of the agent to be described is the high level control and its interaction with the physical platform, therefore it will be described in a separate section later on.

Vehicles Agent sensors provide values resulting directly from the perception they have on the environment. The properties they were designed to provide are: Short range perception (Sonars and infrared detector); Relative location assessment (Altimeter and base direction detector); Self status (Fuel level reader and climb state detector); Long range distance perception (a Device for measuring the minimal and maximal distance to other agents); Environmental conditions (Pollutant reader, pollutant gradient detector); Data communication (Digital radio receiver, landing permission detector); Mission status (Sample density evaluator within a neighbourhood based on communication with the central processing unit); Group status (Gather group detector – radio Signal to follow group).

When dealing with this kind of embodied agent, there are actuators that influence the agents movement and others that actuate over a communication channel. The movement actuators behave by providing the standard helicopter moves: Bank (left and right), Pitch (forward and backward), Yaw (left and right), and Rotor speed (higher or lower). On the other hand there are actuators for data communication with other agents and the processing unit (radio emitter) and for long range presence announcement (ultrasound emitter).

Executor: As this kind of vehicle is very sensitive to actuator and environment changes (v.g. the wind, the rain), thus, it is necessary to ensure that the vehicle stays on a stable position.

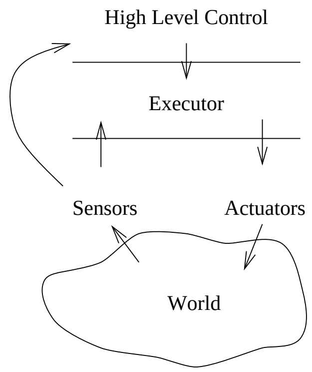


Fig. 3. Executor role in the agent control flow

To meet this goal, the existence of a fast control layer was presupposed. This layer is called *executor* and operates as depicted in Figure 3. The executor is responsible for keeping the referred stable position by monitoring the sensors and operating directly over the actuators within a fast and efficient control cycle. It receives commands from the high level control layer and applies them to the actuators. These commands are described in Figure 4.

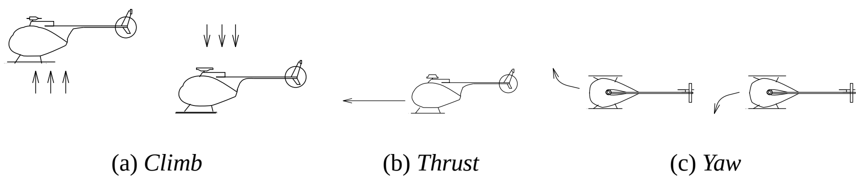


Fig. 4. Movement commands to the executor

Besides movement commands the executor layer implements some other kind of predetermined actions: Sample submission (sends pollutant reading according to pollutant sensor values); Landing base communication (Request permission to land) and Communication with other agents (Call other agents).

3 Agent "Intelligence"

The structure that commands the executor works by receiving information from sensors and sending decisions to actuators. As referred earlier, the approach taken is composed by a set of behaviours in a dynamic decision making architecture [6]. Each behaviour has two outputs: a decision signal and an activity signal. This activity signal varies from null activity to a fixed maximum according to the sensor stimuli and the fatigue/recovery mechanism. The decision is passed to the actuators by comparing the activity output from each pair of conflicting behaviours on a blocker component. This kind of architecture seems to be more suited to unstructured environments in opposition to planning approaches (classic, reactive or opportunistic [10, 1]).

The agent high level control module is a programmable structure that is composed by the referred set of behaviours. This set of behaviours is built according to a programming method. The basic tasks assigned to this method are: information distribution; mission decomposition into a sequence of operations; individual behaviours design and category assignment.

The first step (information distribution) derives from the agent architecture. All the information that an agent possesses is built from its sensor values or given by outside entities that communicate with it. In this case, the values transmitted by outside entities are related to the number of samples taken in the agent neighbourhood as well as the "will" of some other agent to gather a group around it.

In the second step, a sequence of operations is produced. It includes all the major actions that the agent is intended to perform. The sequence obtained for this mission is:

1. Lift off and climb to a safe altitude;
2. Navigate around the pollutant cloud in an "intelligent" way;
3. Take samples according to the sensed pollutant;
4. Refuel and proceed as many times as needed;
5. Return home and land.

This sequence provides the necessary information for the next step. It guides the design of individual behaviours to fulfill all the agents' operations. Finally, these behaviours are assigned to a set of previously defined categories (survival, navigation, mission and group). This result is presented in the next section.

Each one of these categories correspond to a priority level which is the maximum value of the behaviours activity output. The survival category is the most important, the group category is the least important and the other two categories, navigation and mission, are of medium importance.

3.1 Individual Behaviours

The set of behaviours from survival, navigation and mission categories are essentially of individual nature. With these behaviours, an agent should be able to accomplish the mission by itself. So, the functionality of each behaviour is such that the interaction among them compose the desired global behaviour.

Survival Behaviours

- Avoid collisions (horizontal direction): this behaviour is responsible for obstacle avoidance. It takes place at the agent's present altitude. Some fixed rules were built in order to allow the agents to respond to all possible sonar snapshots.
- Avoid collisions (vertical direction): it is a complement to the *Avoid collisions (horizontal direction)* functionality using the bottom sonar. The reason of this logical split is the need for a separate control in the landing process.
- Watch fuel level: This behaviour increases the agents "will" to return home and land according to the fuel sensor value by coordinating *Approach base* and *Land* behaviours. If a lower limit is reached, this behaviour asks for landing permission to the landing base.
- Keep altitude range: The purpose of this behaviour is to keep the agent within an acceptable range of altitudes. At the lower level, this behaviour prevents collisions with most common obstacles (v.g. buildings). On the other hand, the upper level prevents agents from going too high and loose radio control[4].

Navigation Behaviours

- Wander: This behaviour influences actuators in a random fashion. However, it obeys to a set of probabilistic rules defined to provide a smooth wander behaviour.
- Approach base: This behaviour tries to guide the agent home. It tries to maximise the *base direction* sensor value by changing the agent direction. If the agent has no landing permission and it is located directly above the landing base (through the infrared detector value), the behaviour guides it away preventing accidents.
- Land: This behaviour depends on the landing permission and on the infrared sensor that detects presence over the landing base. If conditions are met the behaviour lands the vehicle smoothly by inhibiting the *Avoid collisions (vertical direction)* and *Keep altitude range* behaviours.

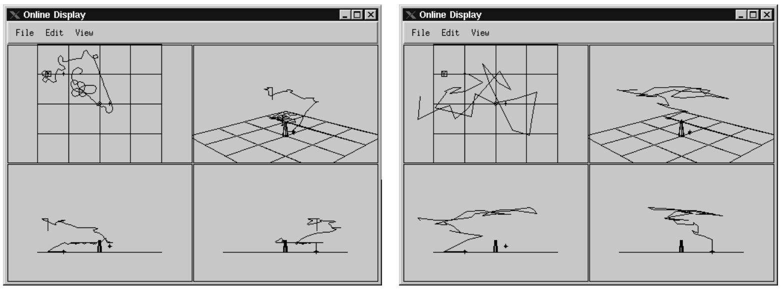
Mission

- Maintain altitude: The cloud reconstruction method as well as the agents stable position are based on the horizontal position. So, this behaviour tries to keep the agent work divided in horizontal layers. The agents manage to go up or down whenever this behaviour decreases its activity level.
- Follow positive gradient: The navigation strategy is composed by two behaviours that follow the pollutant gradient. This one follows positive gradient and the other one, the negative gradient.
- Follow negative gradient: It is complementary to the previous behaviour. The switching of the gradient following method causes the widening of the agent working area and consequently the collecting of a richer variety of samples.
- Collect samples: Pollutant values are discretised in such a way that only significant changes are stored.

[4] Horizontal radio control radius is maintained by the *Approach base* behaviour.

– Avoid over-explored areas: The number of samples taken by the whole society in the present area is given by communication with the processing unit and it can be utilised to decide if this is an interesting area to explore. If that value is too high the agent should go away and explore other areas.
– End mission: Agents should be able to decide when to finish their mission. This behaviour activates the landing procedure after verifying that the number collected samples is equally high for a very long time.

Resulting Behaviour The resulting global behaviour produced by two subsets of the individual behaviours is presented in Figure 5. The survival behaviours implement the lowest level of agents purpose: "being alive". The *Avoid collisions (horizontal and vertical directions)*, *Watch fuel level* and *Keep altitude range* behaviours, keep the agent out of trouble and within an safe working altitude. The navigation behaviours (*Wander*, *Approach base* and *Land*) complement this first global behaviour providing some tools towards an interesting "life time" in the environment. The kind of a combined trajectory (navigation and survival) that can be obtained is shown on Figure 5(a).



(a) Survival and random trajectory (b) Pollutant navigation trajectory

Fig. 5. One agent trajectories

On the other hand, the mission behaviours (*Maintain altitude*, *Follow positive gradient*, *Follow negative gradient*, *Collect Samples*, *Avoid explored areas* and *End mission*) provide the agent the "intelligence" necessary to analyse and monitor the pollutant cloud. The resulting trajectories are of a "come and go" kind, from and to the chimney in order to explore the cloud from the centre to the edges and back[5] (Figure 5(b)).

[5] This was the way considered to be the best for this kind of cloud.

3.2 Group Behaviours

An agent by itself would take a long time to get a reasonable sampling of the cloud. Therefore, group behaviours are designed to increase the society performance by influencing the agents navigation.

The group behaviours' design was based on the group strategies presented in [11] where several kinds of minimal cooperation based on simple communication protocol are used.

There are three major incremental group strategies which can be used separately. The simplest one is a *non-interference* strategy based on the known minimum distance to other agents (direct sensor reading). This assures a sparse distribution of agents in the search space. The second is a *follow* strategy that allows an agent to approach others in order to search for pollutant without external clues (assuming that other agents are inside the cloud). The most complex strategy is intended to improve other agents efficiency by calling them to interesting places (pollutant high level). This is based on both explicit communication and the *follow* feature.

The base behaviours are *Keep minimum distance* and *Keep maximum distance*. The maximising (minimising) procedure for the *Keep minimum distance* (*Keep maximum distance*) behaviour is a trial and error method that toggles between left and right yaw movements when the distance increases (decreases).

- Keep minimum distances from other agents: This behaviour tries to maintain other agents outside a defined neighbourhood. This is achieved by maximising to an upper level the minimum distance value. This is a basic behaviour towards social behaviour.
- Keep maximum distances from other agents: This behaviour, on the other hand, tries to maintain the other agents within a broader neighbourhood by minimising the maximum distance value. This behaviour results in a group maintenance strategy within the distance sensor range.
- Follow group: Based on a communication mechanism with the previous behaviour, it leads the agent towards the others. If all agents activate this behaviour at the same time they form a more compact group. The behaviour is activated on the absence of pollutant or a call from another agent invoking this next behaviour.
- Gather group: This behaviour does not change the agent movement but it tries to gather a group around it by sending a gather signal to all the others.

Resulting Behaviour These behaviours maintain the referred distances in a range that is known and modifiable by other behaviours. In this way, other behaviours can influence the agents global group behaviour by incrementing or decrementing the maximum or minimum distances.

The non-interference strategy results directly from the use of *Keep minimum distance* behaviour with a standard goal distance for the minimum distance to maintain. This distance should be the radius for non-interference that the agent should try to maintain. This is a unilateral behaviour but it results rather well in societies because every agent is trying to maintain that distance as minimum. If one agent tries to decrement that distance far beyond the limit all others would run away.

The follow strategy implemented by the *Follow group* behaviour decrements the maximum distance to keep. This results on a unilateral approach behaviour that does not interfere with the previous strategies thus maintaining a constant minimum distance. The gather group strategy has a behaviour called *Gather group* that does not influence the agents movement. However, it does influence other agents group behaviour through a simple communication protocol. All the agents that receive the gather signal and do not have a "good" pollutant reading start their follow procedure and hopefully approach the calling agent. This could fail because they may not be located near that agent and start following some other agent that did not respond to the call.

3.3 Finding a Missing Cloud

All the described behaviours were tuned to work with the laminar jet pollutant model. The monitoring of a more complex pollutant model as the gaussian one requires a slight change on group strategies to find the pollutant cloud. So, one behaviour was designed to accomplish this task. *Spread* is a behaviour that is the opposite of the *Group* behaviour. It keeps the agents as far as they could be[6].

In this way, in the absence of positive pollutant readings, agents form a wide net that wanders through the environment. The first agent to find the pollutant cloud activates its calling signal and the others group around it. A more complex individual strategy can be combined with this one with benefits. However, this kind of strategies are limited by the agents local knowledge.

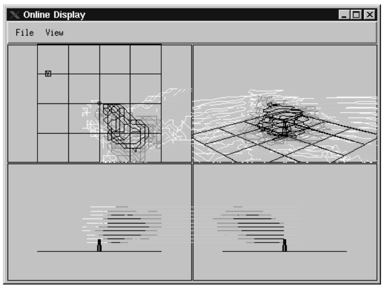


Fig. 6. A cloud obtained with the Spread and Group Strategy

4 Tests and Results

The main goal of this work was to assess group strategies influence on the society performance. This was verified by testing several agent configurations (subsets of the predefined behaviour set) implementing the group strategies described in 3.2. On one

[6] The *Keep Maximum Distances* behaviour ensures that agents stay on the chimney area.

hand, the basic configuration is composed by all individual behaviours, on the other hand, other configurations are implemented according to the group strategies.

The experimented group strategies utilising the laminar jet pollutant model and the mentioned behaviours were: non-interference strategy, follow group strategy, and gather group strategy.

A three agent population simulated for one hour using several behaviour configuration for ten runs each generates a large amount of clouds. These clouds were analysed according to the criterion presented in the next subsection and simple statistical results were obtained. The clouds were built from an average of 300 samples for an average 30000 points grid. In the simulator interface, the collected samples and clouds can be observed in Figure 2.

On the other hand, the tests made with the gaussian pollutant model are limited to finding and monitoring the cloud using a single configuration as referred on the section 3.3. The situation tested represents the worst possible cenarion, that is, the cloud is in the opposite direction to the landing base (Figure 1(b)). The same three agent population was experimented and sixty runs were considered for result analysis.

4.1 Cloud Evaluation

Using the same colour scale applied to visualise the original cloud, on the collected samples, it is possible to compare these two clouds (collected and original) and define an evaluation criterion based on similarity. This evaluation lacks quantification as it is based on a visual assessment. However, the evaluation resulting on *Good* or *Bad* clouds builds a success rate on a range of runs. The criterion for the laminar jet pollutant model is defined by three simple rules. A cloud is considered *Good* if and only if:

1. there are at least three distinct pollutant level values, and
2. there are pollutant readings all around the chimney, and
3. the two highest layers can be considered similar to the original ones (they are in paraboloid shape with approximately the same size).

otherwise the cloud is considered to be *Bad*.

4.2 Group Results

Applying this criterion to the results obtained on all the runs on the referred model, a rate of *Good* clouds is obtained for each agent configuration. Considering the tested configurations after ten runs each the *Good* cloud rates are: for individual behaviours, 40%; for the non-interference strategy, 60%; for the follow group strategy, 65%[7]; and for the gather group strategy, 70%. A significant improvement was detected on using simple group behaviours, just by creating simple synchronism mechanisms (non-interference). This avoids natural work areas overlap between agents configured only with individual behaviours and spreads the agents through the search space. The other strategies improve more subtly the society efficiency by complementing this non-interference

[7] This is a special case where twenty runs were made and considered.

strategy with more coordination mechanisms. The follow group strategy controls the agents spread keeping them within a limited range and making possible for one agent to change its mission status by following others. The gather group strategy enforces this situation by calling agents to better places.

The results obtained for the gaussian model indicate a 85% success rate, that is to say that agents do find and monitor the cloud within a reasonable amount of time.

5 Related Work

There are two areas that are related to this work, that is environmental monitoring and mobile robotics. This work tries to build a bridge between these two areas. In the pollutant analysis Ferreira [8] studied the applications of pollutant indirect measuring by grabbing images from the same chimney that this work intended to simulate.

On the other hand, robotic agents started from the paradigm as it is described in [16], and implemented on a behavioural model inspired by [6, 3, 5, 12, 13] and social interactions were based on [2, 7, 9, 11, 15].

6 Conclusions and Future Work

Pollutant monitoring based on robotic agents improved significantly when relying upon group strategies. All strategies implement a better search method than the previous one, spreading and coordinating all agents through the search space.

Individual behaviours that were designed to search and sample the environment do not guarantee that agents do not overlap work regions or even find the pollutant cloud. With the use of the non-interference strategy agents decrease the number of overlapping situations. With a unilateral follow group behaviour, agents manage to find the cloud assuming others have already found it. At last, the gather group strategy tries to improve even more the global efficiency.

The use of the gaussian pollutant model brings more complex problems. The influence of environment conditions on this model brings a more realistic assessment of this application. The group strategy used for finding the pollutant cloud is obviously sensible to the number of agents utilised. So, its efficiency is expected to increase by adding more agents to the society.

Alternatively, individual strategies may be designed to complement the previous one. However, this kind of strategy is highly dependent on real implementations and available technology. As a future goal, other finding and searching strategies should be experimented and their results compared.

All these promissing results stress out that is possible to perform complex tasks based on the interaction of simple individual and group behaviours based on simple sensor readings.

References

1. Philip E. Agre and David Chapman. Pengi: An implementation of a theory of activity. In *Proceedings of the Sixth National Conference on Artificial Intelligence*, page 268, Seattle, 1987.
2. Ronald C. Arkin and J. David Hobbs. Dimensions of communication and social organization in multi-agent robotic systems. In Meyer et al. [14], page 486.
3. Rodney A. Brooks. A robust layered control system for a mobile robot. *IEEE Journal of robotics and automation*, RA-2(1):14–23, March 1986.
4. Rodney A. Brooks. Intelligence without reason. Technical Report AIM-1293, MIT, April 1991.
5. Jonnathan Connel. *Minimalist Mobile Robotics - perspectives in Artificial Intelligence*. Academic Press, 1990.
6. Luís Correia. Veículos autónomos baseados em comportamentos - um modelo de controlo de decisão. PhD thesis, Universidade Nova de Lisboa, Faculdade de Ciências e Tecnologia, 1995.
7. Peter de Bourcier and Michael Wheeler. Signalling and territorial agression. In Dave Cliff, Phillip Husbands, Jean-Arcady Meyer, and Stewart W. Wilson, editors, *From animals to animats 3 - Proceedings of the third international conference on simulation of adaptive behavior*, page 463. MIT Press, 1994.
8. Francisco C. Ferreira and António S. Câmara. Digital video air pollution monitoring. In *Proceedings of the 15th International Conference in Optical Sensing and Environmental Process Monitoring*, Dallas, 1996.
9. S. Goss and J. L. Deneubourg. Harvesting by a group of robots. In Francisco Varela and Paul Bourgine, editors, *Towards a Practice of Autonomous Systems - Proceedings of the First European Conference on Artificial Intelligence*, page 195, Paris, 1991. MIT Press.
10. Karen Zita Haigh and Manuela M. Veloso. High-level planning and low-level execution: Towards a complete robotic agent. In *Proceedings of the First International Conference on Autonomous Agents*, Marina del Rey, CA, February 1997. Available in `http://www.cs.cmu.edu/˜mmv/produce-bib.html`.
11. C. Ronald Kube and Hong Zhang. Collective robotic intelligence. In Meyer et al. [14], page 460.
12. Pattie Maes. A bottom-up mechanism behavior selection in an artificial creature. In Jean-Arcady Meyer and Stewart W. Wilson, editors, *From animals to animats - Proceedings of the Adaptive Behavior Conference*, page 238. MIT Press, 1990.
13. Maja J. Matarić. Designing emergent behaviors: From local interactions to collective intelligence. In Meyer et al. [14], page 432.
14. Jean-Arcady Meyer, Herbert L. Roitblat, and Stewart W. Wilson, editors. MIT Press, 1992.
15. Adam Walker and Michael Wooldridge. Understanding the emergence of conventions in multi-agents systems, June 1995. Disponível em `http://www.dlib.com/people/mjw/papers.html`.
16. Michael Wooldridge and Nicholas R. Jennings. Agent theories, architectures, and languages: A survey. In Michael Wooldridge and Nicholas R. Jennings, editors, *Intelligent Agents - Proceedings of the Eleventh European Conference on Artificial Intelligence (ECAI-94)*, page 1, Amesterdam, August 1994. Springer-Verlag.
17. Kenneth Kuan yun Kuo. *Principles of Combustion*. John Wiley & Sons, New York, 1987.