A Society of Agents in Environmental Monitoring

João Costa Seco[*]
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@di.fct.unl.pt

Carlos Pinto-Ferreira
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

Luís Correia
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
lc@di.fct.unl.pt

January 19, 1998

Keywords: Agents, Agent societies, Autonomous robots, Simulation models, Pollutant monitoring

Abstract

The evaluation of pollutant levels is a key aspect on the issue of keeping a clean environment. Conventional techniques include the utilisation of a fixed setup incorporating pollutant sensors. However, these approaches are a very long way from an accurate monitoring. Thus, to improve pollutant monitoring on a power plant chimney, the use of robotic agent societies (mobile robots) is suggested. This suggestion is adequate in pollutant monitoring when the environment is hostile and/or the region to be sampled has large dimensions.

However, the implementation of a system incorporating robotic agents raises complex technological problems. Before a set of any kind of real robotic agents is implemented, an accurate evaluation must be performed. What this paper describes is a simulation of an application of small flying robotic agent societies (helicopter models) monitoring a pollutant cloud.

This simulation intends to show that an "intelligent" search method works better than a systematic or random procedure. In this kind of environment (dynamic and non-structured) and using mobile robotics to meet a goal such as this, a behavioural control architecture seems to meet the performance objectives.

The behaviours designed to control the agents are prepared to implement individual needs (survival and navigation) and social needs (follow or gather group). The agents as individuals are capable of performing such a mission, however, global results are enhanced by social strategies.

Topic areas: Evaluation of robot/simulation models, Collective and social behaviour, Autonomous robots.

This paper is intended to be a long paper.

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

1 Introduction

The simulation of a robotic agent society monitoring a pollutant cloud, originated from a power plant chimney, is the major issue on this work and was the motivation for a project proposal [Seco, 1997] dedicated to reactive agent societies.

These agents have a simple mission: to go around a chimney and sample the pollutant cloud building a global map on a central processing unit on the ground. This map is transformed into an image that holds information about cloud direction, pollutant concentration, etc., allowing 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. The interpolation process 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 [Correia, 1995]. This kind of architecture is designed to deal with multiple sensors, multiple goals and be robust in non-structured environments [Brooks, 1986, Brooks, 1991].

To assess the efficiency of the social behaviour, a simulator creates a modelled pollutant cloud, launches a set of robotic agents the sample that cloud using several group behaviour strategies and the collected clouds are compared to the one that was 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 [Kube and Zhang, 1992] on minimal group behaviours. These concerns a non-interference strategy as a base for more complex behaviours such as follow or gather.

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 atention on behaviours design.

The whole system is composed by a few 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 produce actions based on sensor readings and the processing unit produces clouds based on samples taken by the agents.

2 Agent Society Description

2.1 The Proposed Architecture

In this application, agents are small helicopter 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 a social behaviour adequate to the mission goals.

There were options taken in all these topics. What the next section explains is the context in which they were made. The perception and actuation are the basic part of the agent and need to be described in detail. The other important part of the agent is the agents programming and its interaction with the previous parts.

2.1.1 The Vehicles

Agent sensors provide values resulting directly from the perception they have on the environment. The properties they were designed to provide are:

1. Short range perception: Sonars, infrared detector;
2. Relative location assessment: Altimeter, base direction detector;
3. Self status: Fuel level reader, climb state detector;

4. Long range distance perception: Device for measuring the minimal and maximal distance to other agents based on a ultrasound receiver;
5. Environmental conditions: Pollutant reader, pollutant gradient detector;
6. Data communication: Digital radio receiver, landing permission detector;
7. Mission status: Sample density evaluator within a neighbourhood (based on a communication with the central processing unit);
8. Group status: Gather group detector (radio signal to follow group);

When dealing with an embodied agent, there are actuators that influence the agents movement and others that actuate over a communication channel. The movement actuators behave by giving 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 the communication actuators are the following:

1. Data communication with other agents and the processing unit: Radio emitter.
2. Long range presence announcement: Ultrasound emitter.

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

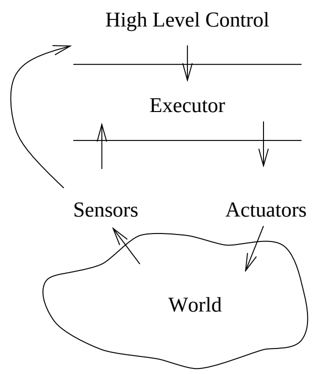
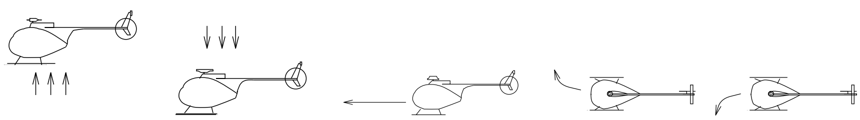


Figure 1: Executor role in the agent control flow

To meet this goal, the existence of a fast control layer was presupposed. This layer is called *executor* and operates as described in Figure 1.

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 2.



(a) *Climb* (b) *Thrust* (c) *Yaw*

Figure 2: Movement commands to the executor

Besides movement commands the executor layer implements some other kind of predetermined actions:

1. Sample submission: Sends pollutant reading according to pollutant sensor values.
2. Landing base communication: Request permission to land.
3. Communication with other agents: Call other agents.

2.1.2 The 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.

Added to 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 methods and the global cloud is calculated by gathering the samples from all agents and performing data interpolation.

2.2 Programming

The structure that commands the executor works by receiving information from sensors and sending decisions to actuators. The approach taken is composed by a set of behaviours in a dynamic decision making architecture [Correia, 1995]. Each behaviour has two outputs: an action signal and an activity signal. This activity signal varies from null activity to a fixed maximum according to the sensor stimuli. The decision is made comparing the activity output from each pair of conflicting behaviours. This kind of architecture seems to be more suited to unstructured environments in opposition to planning (classic, reactive or opportunistic) [Haigh and Veloso, 1997, Agre and Chapman, 1987].

2.2.1 Agent Programming

The agent high level control module is a programmable structure that is composed by a 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 do. 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 designs 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. 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.

To implement this method, mechanisms for coordination and communication between behaviours were aplied. These mechanisms allow the implementation of a structured behaviour set.

2.2.2 A Set of Minimal Behaviours

The identified priority classes and the corresponding behaviours are as follows:

1. Survival behaviours:
   (a) Avoid collisions (horizontal direction);
   (b) Avoid collisions (vertical direction);
   (c) Watch fuel level;
   (d) Keep altitude range;

2. Navigation behaviours:
   (a) Wander;
   (b) Approach base;
   (c) Land;
3. Mission behaviours:
   (a) Maintain altitude;
   (b) Follow positive gradient;
   (c) Follow negative gradient;
   (d) Collect samples;
   (e) Avoid over-explored areas;
   (f) End mission;
4. Group behaviours:
   (a) Keep minimum distance from other agents;
   (b) Keep maximum distance from other agents;
   (c) Follow group;
   (d) Gather group;

The functionality of each behaviour is such that their interaction fulfills the desired global behaviour. It is as follows:

1. Avoid collisions (horizontal direction): this behaviour is responsible for obstacle avoidance. It takes place at the agent present altitude. Some fixed rules were built in order to allow the agents to respond to all possible sonar snapshots.
2. 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.
3. 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.
4. 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[1].
5. Wander: This behaviour influences actuators in a random fashion. However, this behaviour, however, obeys to a set of probabilistic rules defined to provide a smooth wander behaviour.
6. 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.
7. 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.
8. 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 when this behaviour ceases its activity.
9. 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.

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

10. 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.

11. Collect samples: Pollutant values are discretised in such a way that only significant changes are stored.

12. Avoid over-explored areas: The number of samples taken by the whole society in the present area is given by a sensor and it can be used 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.

13. 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.

14. 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.

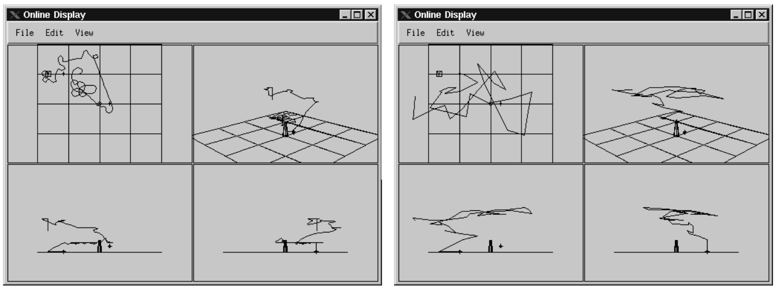
15. Keep maximum distances from other agents: This behaviour, on the other hand, tries to maintain the other agents within a broader neighbourhood by minimising the maximum distance value. This behaviour results in a group maintenance strategy within the distance sensor range.

16. 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.

17. 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.

2.2.3 Emergent Individual Behaviour

The behaviour priority classes give the possibility of testing all aspects of the global behaviour.



(a) Survival and random trajectory

(b) Pollutant navigation trajectory

Figure 3: One agent trajectories

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 3(a).

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[2] (Figure 3(b)).

[2]This was the way considered to be the best for this kind of cloud. The analysed cloud is presented in section 3.1

2.2.4 Emergent Group Behaviour

The group behaviours design was based on the group strategies presented in [Kube and Zhang, 1992] 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 the agents in the search space. The second group strategy is a *follow* feature that allows an agent to approach other agents 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).

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 feature is implemented by the *Follow group* behaviour that decrements the maximum distance. 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 Pollutant Modelling

The pollutant processing has two separate procedures, an environment simulation for the original cloud generation and a sample processing and collected cloud generation.

3.1 Original Cloud Modelling

An analytical formula from the laminar jet dispersion theory [yun Kuo, 1987] was used in order to simulate a cloud produced by a power plant chimney. This creates a cloud like the one represented on Figure 4.

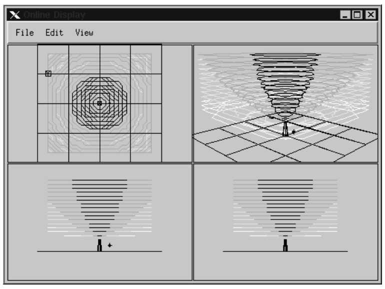


Figure 4: Original pollutant cloud

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.

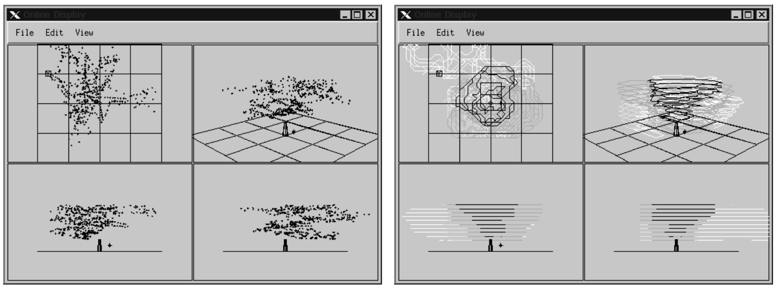
Other simulation models could be used, by simply replacing the pollutant module in the simulator structure.

3.2 Sample Interpolation and Cloud Building Algorithm

All samples are gathered with the information of the pollutant value, sampling time and sample location. With this information it is possible to draw a window on the time axis and follow a possible evolution of the cloud. In the present model this is not feasible due to the static nature of the pollutant model.

The samples are used to calculate a three dimensional grid that contains all the samples taken and has a fixed step[3]. 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[4].

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.



(a) Collected samples

(b) Interpolated cloud

Figure 5: Agent Society Results

4 Results

on 10 runs

There were several agent configuration (a subset of all the predefined behaviours) tested: Individual behaviours, non-intereference strategy, follow group strategy, and gather group strategy. A three agent population simulated for one hour using several behaviour configuration generates a large amount of clouds. These clouds were analysed 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 5.

4.1 Cloud Evaluation

.

Using the same discretization, applied to the original cloud, on the collected samples, it is possible to compare the two clouds 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.

This criterion is defined on three rules. A cloud is considered *Good* 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*.

[3] The value used in this simulation is about 50m.

[4] The six closest points within a range of 200m are considered to be eligible for every point on the grid.

4.2 Group Results

Applying this criterion to the results obtained on all the runs, a percentage of *Good* clouds is obtained for each agent configuration. The behaviour configurations are:

1. Individual behaviours: All the survival and navigation behaviours.
2. Non-interference: The previous configuration plus the *Keep minimum distance* behaviour.
3. Group: The previous configuration plus the *Keep maximum distances* and the *Group* behaviours.
4. Call Group: The previous behaviours plus the *Call Group* behaviour.

and the results are:

1. Individual behaviours: 40% of *Good* Clouds.
2. Non-interference: 60% of *Good* Clouds.
3. Follow Group: 65% of *Good* Clouds.
4. Gather Group: 70% of *Good* Clouds.

5 Related Work

This work builds a bridge between the environmental monitoring and the mobile robotics. In the pollutant analysis Francisco Ferreira [Ferreira and Câmara, 1996] 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 [Wooldridge and Jennings, 1994], and implemented on a behavioural model inspired by [Correia, 1995, Brooks, 1986, Connel, 1990, Maes, 1990, Matarić, 1992] and social interactions were based on [Arkin and Hobbs, 1992, de Bourcier and Wheeler, 1994, Goss and Deneubourg, 1991, Kube and Zhang, 1992, Walker and Wooldridge, 1995].

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.

This pollutant model is not the most accurate because it is not affected by wind or any other environmental conditions. A future step on this work is the utilisation of a new pollutant model that is influenced by all these conditions and the corresponding study on the behaviours design.

References

[Agre and Chapman, 1987] Agre, P. E. and Chapman, D. (1987). Pengi: An implementation of a theory of activity. In *Proceedings of the Sixth National Conference on Artificial Intelligence*, page 268, Seattle.

[Arkin and Hobbs, 1992] Arkin, R. C. and Hobbs, J. D. (1992). Dimensions of communication and social organization in multi-agent robotic systems. In [Meyer et al., 1992], page 486.

[Brooks, 1986] Brooks, R. A. (1986). A robust layered control system for a mobile robot. *IEEE Journal of robotics and automation*, RA-2(1):14–23.

[Brooks, 1991] Brooks, R. A. (1991). Intelligence without reason. Technical Report AIM-1293, MIT.

[Connel, 1990] Connel, J. (1990). *Minimalist Mobile Robotics - perspectives in Artificial Intelligence*. Academic Press.

[Correia, 1995] Correia, L. (1995). Veículos autónomos baseados em comportamentos - um modelo de controlo de decisão. PhD thesis, Universidade Nova de Lisboa, Faculdade de Ciências e Tecnologia.

[de Bourcier and Wheeler, 1994] de Bourcier, P. and Wheeler, M. (1994). Signalling and territorial agression. In Cliff, D., Husbands, P., Meyer, J.-A., and Wilson, S. W., editors, *From animals to animats 3 - Proceedings of the third international conference on simulation of adaptive behavior*, page 463. MIT Press.

[Ferreira and Câmara, 1996] Ferreira, F. C. and Câmara, A. S. (1996). Digital video air pollution monitoring. In *Proceedings of the 15th International Conference in Optical Sensing and Environmental Process Monitoring*, Dallas.

[Goss and Deneubourg, 1991] Goss, S. and Deneubourg, J. L. (1991). Harvesting by a group of robots. In Varela, F. and Bourgine, P., editors, *Towards a Practice of Autonomous Systems - Proceedings of the First European Conference on Artificial Intelligence*, page 195, Paris. MIT Press.

[Haigh and Veloso, 1997] Haigh, K. Z. and Veloso, M. M. (1997). 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. Available in `http://www.cs.cmu.edu/˜mmv/produce-bib.html`.

[Kube and Zhang, 1992] Kube, C. R. and Zhang, H. (1992). Collective robotic intelligence. In [Meyer et al., 1992], page 460.

[Maes, 1990] Maes, P. (1990). A bottom-up mechanism behavior selection in an artificial creature. In Meyer, J.-A. and Wilson, S. W., editors, *From animals to animats - Proceedings of the Adaptive Behavior Conference*, page 238. MIT Press.

[Matarić, 1992] Matarić, M. J. (1992). Designing emergent behaviors: From local interactions to collective intelligence. In [Meyer et al., 1992], page 432.

[Meyer et al., 1992] Meyer, J.-A., Roitblat, H. L., and Wilson, S. W., editors (1992). MIT Press.

[Seco, 1997] Seco, J. C. (1997). Monitorização de nuvens de poluente através de sociedades de agentes robóticos simulados. Master's thesis, Universidade Nova de Lisboa, Faculdade de Ciências e Tecnologia.

[Walker and Wooldridge, 1995] Walker, A. and Wooldridge, M. (1995). Understanding the emergence of conventions in multi-agents systems. Disponível em `http://www.dlib.com/people/mjw/papers.html`.

[Wooldridge and Jennings, 1994] Wooldridge, M. and Jennings, N. R. (1994). Agent theories, architectures, and languages: A survey. In Wooldridge, M. and Jennings, N. R., editors, *Intelligent Agents - Proceedings of the Eleventh European Conference on Artificial Intelligence (ECAI-94)*, page 1, Amesterdam. Springer-Verlag.

[yun Kuo, 1987] yun Kuo, K. K. (1987). *Principles of Combustion*. John Wiley & Sons, New York.

Contents