Dynamic Spill Monitoring: A Coordinated Approach

Peter Jones

James K. Archibald
james_archibald@byu.edu

See next page for additional authors

Follow this and additional works at: https://scholarsarchive.byu.edu/facpub

Part of the Electrical and Computer Engineering Commons

BYU ScholarsArchive Citation
All Faculty Publications. 1321.
https://scholarsarchive.byu.edu/facpub/1321

This Report is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in All Faculty Publications by an authorized administrator of BYU ScholarsArchive. For more information, please contact scholarsarchive@byu.edu, ellen_amatangelo@byu.edu.
Abstract

This paper presents a coordinated control strategy for autonomous monitoring of a dynamically changing hazardous spill. The control strategy is a two level hierarchy that uses coordination variables to communicate between the levels. The upper level is a resource allocator. The lower level consists of several “commanders”, each with a possibly different objective. A physical description of the problem is given, as well as the methods of spill modeling and simulation. Tactics for locating, gravitating to and circling the spill are presented. Finally, both simulation and hardware results are given.

1 Introduction and Motivation

Monitoring the spread and diffusion of hazardous wastes into the environment is an extremely important problem. Remote sensing of waste or pollutants using cameras or radar is not well suited to many types of hazardous materials like, for example, radioactive wastes. An alternative is to place multiple physical sensors in the environment. However, if these sensors are statically allocated, they will be unable to adjust to a dynamically varying environment. Locating these sensors on mobile, autonomous vehicles facilitates the movement of sensors to locations where the hazardous material is moving the quickest.

The possibility of using multiple, mobile robots for jobs that are either too difficult or too hazardous for humans has already been discussed in several papers [1, 2, 3]. In [4] and [5], robots have been specifically designed for the monitoring and containment of hazardous waste. As such robots become increasingly inexpensive and the benefits of using autonomous agents for hazardous tasks become more obvious, designers must find increasingly efficient methods for coordinating the efforts of these agents. Such designs must implement not only methods of efficient, structured inter-agent communication, but also allow an external user to interface with the robot group. Additionally, users should be able to issue a wide range of commands both to robot groups and to individual agents.

In previous papers, the concept of formation feedback through the use of coordination variables has been explored [6, 7]. In this paper we describe a variation of the formation feedback design, specifically adapted to the task of monitoring a dynamic hazardous waste spill. We focus on a two-dimensional case (such as waste leaking from a barrel or ground-level pipe), but the work could easily be extended into a three-dimensional spill, such as an atmospheric particle plume or an oceanic oil spill.

Our current implementation uses a double formation feedback architecture. In this architecture, each agent communicates with a commander. The commander then broadcasts a coordination variable to all agents under its command. Additionally, each commander communicates with an overseeing arbiter, which coordinates the efforts of the separate commanders and is responsible for agent group assignment.

The rest of the paper will be organized as follows. In section 2 we briefly review the concept of formation feedback, its benefits and limitations, and its suitability for the problem at hand. In section 3 we describe the agents’ world, including spill modelling and sensory assumptions. In section 4 we will address the need for human control and the implementation of that control. In section 5 we present both simulation and hardware results.

2 Coordination Architecture

According to one author, “The question of communication is fundamental to the problem of cooperation: How, what, and when should robots communicate to achieve a given task?” [8] Several architectures, each with its individual communication paradigm, have been introduced for Multi-Robot Cooperation (an interesting comparison of MRC architectures can be
found in [9].) Our architecture borrows from several of these proposed organizations while using the coordination variable communication paradigm initially developed for the purpose of formation maneuvers [6]. We show that this paradigm can be readily extended to more general applications.

2.1 Formation Feedback

Much of the literature in the area of multiple robot formation keeping is concentrated in two basic strategies. In the leader-following architecture [10, 11, 12] the leader tracks a pre-specified trajectory and the followers track a transformed version of the leader’s states. The advantage of such a strategy is that formation maneuvers can be completely specified in terms of a leader’s trajectory. One of the disadvantages is that there is no inherent feedback from the followers to the leader. Therefore a failure to maintain formation on the part of the followers won’t affect the leader’s trajectory or the behavior of the other followers unless group feedback is explicitly included in the design.

A second approach to formation keeping is a behavioral strategy [13, 14, 15]. In this approach a number of individual responses to local conditions (or behaviors) are specified for each robot. In this manner formation feedback is implicitly contained in the low level decisions of the robots. Unfortunately, such an approach makes it very difficult to ensure group stability or to completely specify a complex group behavior.

It has been suggested that treating the robot formation as a “virtual structure” would combine the relative advantages of leader-following and behavior based strategies. In [16] it was demonstrated that the virtual structure scheme results in a system that includes formation feedback, can easily prescribe formation maneuvers, and has guaranteed formation stability.

Figure 1 shows an architecture proposed in [16] for implementing a virtual structure approach to formation keeping. In this architecture, the supervisor makes high level decisions based on user input. The Formation Control block accepts as inputs the Supervisor’s statement of overall mission objective as well as state information from each robot and sends out a Coordination Variable, which communicates to the individual agents all the external information they need to know in order to cooperate.

2.2 Current Implementation

The problem of monitoring a dynamically expanding spill presents an interesting application of the coordination variable architecture. Because of the non-static nature of the spill, a simple virtual structure approach is overly constrictive. However, the coordination variable concept can be generalized to the problem by letting \( \xi \) represent the minimal amount of information needed to facilitate cooperative behavior. An additional advantage of the architecture is the ease with which it can be extended to a multiple level implementation, as demonstrated in the system architecture employed in our spill monitoring system.

An extended architecture is shown in Figure 2. This shows multiple entities transmitting Coordination Variables each to a specific grouping of the system. Each commander is responsible for a specific task in the spill world. When needed, the commanders request additional resources from the arbiter, who then allocates robots to the commanders, depending on cost and benefit functions. Here again, the information can be considered as a higher level coordination variable.

When the monitoring task begins, there is a single commander who is responsible for searching the area for dangerously high levels of waste. Initially, all robots belong to the search commander. The commander coordinates the movements of the robots in a sweeping search until one robot’s sensors pick up a sufficiently high waste level. When it is determined that a new spill exists, a new commander is instantiated to deal with that particular spill. The spill comman-
The commander is responsible for evaluating its need for additional resources and requesting those resources from the arbiter, as well as coordinating the efforts of the agents allocated to it.

The two commanders (search and spill #1) have separate goals, so \( G \) must act as arbiter, allocating resources based on perceived costs and benefits. The coordination variable \( \xi_1 \) commands a change of robot/commander assignment. This limits the scope of the commanders jobs to simply using the robots they have, rather than fighting over resources.

The structure of the coordination variable \( \xi_{12} \), issued by the commanders to their robots, are subtask dependent. For example, the search commander sends x-y goal coordinates to each of the robots. This allows a more flexible search routine than a rigid structure. However, if minimal flexibility were desired, it could form the robots under its command into a virtual structure, and then communicate translation, rotation and expansion commands. In the case of spill commanders, \( \xi_{12} \) coordinates which sections of the spill are patrolled by which robots. Each robot is allocated a different section of the spill based upon spill boundary length and rate of diffusion across the boundary. In this way the areas which are growing most quickly are monitored more frequently, allowing better overall monitoring of the spill.

In addition to obstacle avoidance, the low level control function implements three primitive behaviors. The first is a target following behavior, used in formation keeping, the second is triggered by encountering a level of waste above a threshold value and consists of a gradient ascent, the third is a patrolling behavior. When the robot senses a level of material above a lower threshold, it begins to ascend the gradient. As it approaches an upper threshold or patrolling level, it smoothly transitions into a patrolling behavior. This is accomplished by using a sigmoid weighting function:

\[
W(s) = \frac{1}{1 + \exp(-\sigma(s/s_m - .5))}
\]

This function depends on the current sensor reading \( s \) and the constants \( s_m \) (upper threshold concentration) and \( \sigma \) which controls the abruptness of the transition.

## 3 World Description

One of the difficulties in building and testing a reliable architecture to monitor a hazardous waste spill is in modelling the spill. Because of the impracticability of using actual spills and sensors, computer models need to be employed.

In this work, hazardous spills are modelled using Matlab’s partial differential equation (pde) toolbox. Diffusion can be modelled using the parabolic pde:

\[
du' = f + c\nabla^2(u)
\]

Where \( u \) is the material concentration, \( c \) is the material’s diffusivity, etc. Matlab enables the user to specify fairly complicated boundaries over which to allow the spill to diffuse, as well as controlling boundary values (both Neumann and Dirichlet conditions.) This tool enabled accurate modeling of a two dimensional spill under a large set of variable conditions such as room size and geometry, material diffusivity, initial concentration, rate of material infusion, etc.

In addition to the spill, hazardous material sensors were also simulated (all other hardware and sensors are real). We have assumed four on-board sensors (at 0, 90, 180 and 270 degrees), each of which can sense concentration variations of approximately 1% of the initial spill concentration.

Brigham Young University’s Multi-AGent Intelligent Coordinated Control (MAGICC) lab uses a 15’ x 15’ test bed with a team of five homogenous custom built mobil robots. Absolute position information is obtained via an overhead camera at a rate of approximately 50 frames per second. The robots have a PC104 stack on board and communicate with host PCs over a 10 Mb/s wireless LAN.

The simulations are run using the Matlab/Simulink Real Time Workshop software. Additionally, a
Simulink toolbox has been developed in-house [17] to help with position acquisition and robot motor control as well as robot/pc communication.

4 Human Interface

The critical nature of monitoring and controlling a hazardous waste spill indicates the need for a human interface. For the interface to be effective, the user must be able to control both specific robots as well as robot groups. The human commander will also be competing with the autonomous commanders for robot resources. This introduces the challenges inherent in mixed-initiative systems with adjustable autonomy.

Implementations of mixed initiative systems have been discussed in-depth in [18, 19]. The simplicity of our system allows for a much more rudimentary solution than those presented in the literature. Adjustable or adaptive autonomy has also been discussed at length in [20, 21, 22]. Again, principles have been applied to this project in a very direct and specific fashion which avoids several of the difficulties of more general implementations.

An additional concern in the design of a human interface is the method of information dissemination. Sensor information must be presented to the user in such a way that he or she can quickly evaluate it. Additionally, the user must be able to quickly and easily vary his or her requests for agents or alter the goal of the controlled group. All these interface requirements are accentuated in the case of a high pressure, time-constrained operation, such as the one for which this system has been designed.

4.1 Mixed Initiative

Any system in which limited resources must be used to accomplish multiple goals simultaneously, necessarily demonstrates an element of mixed initiatives. In our system, each commander module has a need for agents either to search or to patrol. Additionally, a human interfacing with the software system may also request certain resources. Each commander module and each human user has a different goal, so the arbiter must act to determine which entity controls the initiative for each agent.

The method of determining which goal each agent will pursue (i.e., determining who has initiative) involves not only an assessment of each commander/user’s need, but also an evaluation of each agent’s ability to fill that need. Referring to figure 2, the arbiter accepts $z_{F_i}$ from each commander, as well as the human interface. The arbiter then computes the ability of each agent to fill each commanders’ needs (this is similar to the concept of impatience in the alliance architecture [3]). Currently this ability is based solely on position, but could be extended to include heterogeneous aspects of each robots such as specific sensors or actuators. Once the arbiter has decided which entity has initiative over which agents, the information is broadcast via the coordination variable $\xi_1$ which is received by each commander and rendered visually to the user.

The constrained environment we’ve considered in the development of this project and the limited set of actions the robots may be involved in eliminate several of the difficulties inherent in most mixed initiative systems. For a more complete discussion of the challenges and promises of mixed initiative, the reader is directed to [23].

4.2 Adjustable Autonomy

An additional difficulty in having a human commander interface with a multi-agent system is the need to control a variable number of robots. In [24] an implementation of human control of multiple robots was demonstrated. This system enables a user to control a single or several robots and to issue either specific or general goals.

This type of adjustable autonomous behavior is very attractive for the scenario of hazardous waste monitoring. A human user may wish to have explicit control of a specific robot or group in order to achieve a complicated task. Or a user may only wish to give a robot or group general commands to search a previously unexplored region. Thus the system must generate an adjustable autonomy on the part of the agents controlled by the human commander, possibly ranging from direct teleoperation to relatively autonomous searching with nothing more than a region specified.

4.3 User Interface

Figure 3 shows a simulated human command interface to the hazmat system. Several principles of user interface design were considered while creating this simulated command center.

- Consistency: The issuing of commands and categorizing of robots occur in a self-consistent manner.
Visibility/Affordances: A color-coded map shows both robots’ paths and sensor values. Robots’ current positions and groupings are delineated in the left-hand portion of the GUI.

Feedback: The results of a user’s actions are immediately visible to them in the form of the map.

Control

Direct Manipulation

User Information Load

Burden of Interpretation

5 Results

5.1 Simulation

The architecture has been largely implemented in simulation, and tested for one, two and five agents. The following plots demonstrate the five robot case when there are multiple spills. Notice that each robot assigned to a spill receives an area to patrol, communicated to the agent from the spill commander (this is the essential information communicated in the coordination variable $\xi_2$.) When the overseeing arbiter reconfigures agent/commander configurations, these patrol areas are altered. By assigning robots to patrol specific sections of the spill or spills, we’ve created a dynamic distribution such that spill areas which are expanding more quickly will be monitored more closely.

5.2 Hardware

The MMRT toolbox described in [17] enables a flawless transition from simulation to hardware implementation. The system was tested for one and two robot systems, and will shortly be implemented in the five robot case.

Acknowledgment

This paper was funded by the Idaho National Engineering and Environmental Laboratory under contract #?. The authors especially thank Mr. Don Duddenhoeffer for his technical guidance of the project.
References


