Robotics and Autonomous Systems 62 (2014) 91–107

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot

Towards a robust feedback system for coordinating a hierarchical multi-robot system

Praneel Chand [a,b,*], Dale A. Carnegie [b]

[a] School of Engineering and Physics, The University of the South Pacific, Suva, Fiji

[b] School of Engineering and Computer Science, Victoria University of Wellington, New Zealand

HIGHLIGHTS

- A feedback coordination system for a hierarchical heterogeneous robot team is presented.
- The system is designed to be used with a reduced human user input task allocation system.
- Poor performance, partial failures, and complete failures are detected and corrected.
- The system is robust to threshold variation and monitor time interval variation within the tested limits.

ARTICLE INFO

Article history:
Received 11 March 2013
Received in revised form
14 November 2013
Accepted 22 November 2013
Available online 7 December 2013

Keywords:
Multi-robot systems
Hierarchical
Heterogeneous
Limited capability robots
Feedback coordination

ABSTRACT

Restricting the usage of a team of robots to a few expert human users can be disadvantageous. In applications such as exploration, it may not always be possible for human experts to travel to sites, resulting in negative consequences. It is preferable to have a robotic system that is capable of coordinating itself based on inputs provided by non-expert human users. Hence, this paper presents the development of a robust feedback system for coordinating a hierarchical team of robots where inputs are specified by non-expert human users. Experiments with a multi-robot mapping and exploration task show that the feedback system successfully detects and corrects three types of failures. These are poor performance, partial failure and complete failure. Moreover, the system is robust to threshold value variation and monitor time interval variation within the tested limits.

1. Introduction

Cooperative behaviour in multi-robot systems can result in several advantages. Efficiency can be increased and task completion times can be reduced when multiple robots work in parallel. Redundancy can be introduced to improve reliability. Smaller simplistic machine designs can reduce manufacturing costs in multi-robot systems.

Central to the success of many multi-robot systems are task allocation and coordination mechanisms. Task allocation mechanisms distribute tasks between different robots [1]. Coordination mechanisms allow individual robots within a group to take each other's actions into consideration such that the team operates coherently [2].

A range of task allocation methods have been reviewed in [1,3–5]. Many of these methods utilise detailed expert knowledge to coordinate robots. This often means that an expert human user is required to initialise the robots. However, non-expert human users may be required when it is not possible for human expert users to physically travel quickly to exploration or search and rescue sites.

Recently, a reduced human user input task allocation method [5] has been proposed. The main purpose of the approach is to enable non-expert human users to specify inputs to multi-robot systems. This is achieved by employing fuzzy systems [6] to reduce the quantity and range of input values needed. Tasks are specified in terms of four broad categories of robot hardware resources (processing, communication, sensing and actuation). The task allocation method is well suited for hierarchical heterogeneous systems such as [7]. A hybrid reactive–deliberative control architecture [8,9] is utilised in this multi-robot system.

Using non-expert human user inputs for task allocation has a limitation. The reduced inputs provided by the non-expert user can potentially be incorrect due to human error. This can produce incorrect task–robot matching leading to robot failures during task

* Corresponding author at: School of Engineering and Physics, The University of the South Pacific, Suva, Fiji. Tel.: +679 9099749.
E-mail addresses: chand_pc@usp.ac.fj (P. Chand), dale.carnegie@vuw.ac.nz (D.A. Carnegie).

0921-8890/$ – see front matter © 2013 Published by Elsevier B.V.
http://dx.doi.org/10.1016/j.robot.2013.11.010

execution. Additionally, other robot failures may arise due to unexpected hardware failure or interactions with the environment. Some form of feedback is required to address such failures.

A feedback system [10] has been implemented and tested with a task allocation system that utilises detailed numerical data [11]. Similar to task specification, four main categories of robot hardware (processing, communication, sensing and actuation) are monitored. However, the results presented in [10] were preliminary. Additionally, the system has not been evaluated for reduced human user input task allocation. Hence, this paper presents the development of a robust feedback system for coordinating a hierarchical heterogeneous team of robots where reduced human user input task allocation is utilised.

2. Related work

Several task allocation and coordination strategies for multi-robot systems have been discussed in [1–5]. Whereas [1,3–5] focus on task allocation, [2] addresses coordination. Task allocation methods have also been utilised in coalition formation [12–14] to enable multiple robots to collectively achieve the objectives of a task that individual robots are incapable of executing.

Fault tolerance in multi-robot systems can be viewed as a specialised form of multi-robot coordination. Some form of feedback is employed during the execution of a task to detect robot failures. This feedback can be in the form of performance metrics and monitors. When failures are detected, the robotic team responds by dynamically reselecting or reassigning the tasks of failed robots.

Parker [15] has implemented performance monitors for behaviour sets in the L-ALLIANCE architecture. Task completion time is used as the performance metric in this architecture. Task reallocation is effected via a learning process that updates the control parameters of the behaviour sets associated with the task that a robot executes. A drawback of L-ALLIANCE is that it is tailored for behaviour based systems making it potentially difficult to apply to non-behaviour based architectures.

Dias et al. [16] address the issue of robustness in dynamic environments by categorising three forms of robot malfunctions. These are complete robot failure, partial robot malfunction, and communication failures. Experiments with a homogeneous set of three robots performing a distributed sensing problem evaluate the detection of failures. The system (TraderBots) permits graceful degradation of team performance when failures occur. In this approach, all available robots are deployed for a task and robots are not replaced when they fail.

Kannan and Parker [17] have developed task execution success and failure metrics to investigate the influence of fault tolerance on overall system performance. In their implementation, the robots are required to perform a number of tasks and each robot–task pair contributes towards the overall performance. The overall performance is the difference in the reward gained from successfully executed tasks (success metric) and the punishment for unsuccessful task execution (failure metric). A drawback of this approach is that performance is only determined after task execution completes and not during task execution.

An extension to Kannan and Parker's work on fault tolerance [18] measures the effectiveness of fault tolerance in box pushing and deployment tasks. Fault tolerance in these tasks is tested using predefined and adaptive causal model methods. However, the implementation of causal model methods can be cumbersome when there are many robots, fault nodes, and fault combinations. Additionally, causal model methods need to be tailored for the task that the robots execute and the environment that they operate in.

Tolerance to sensor failures in a small team of distributed robots has been investigated in [19]. This research extends the sensing fault tolerance capability of the Sensor Fusion Effects (SFX-EH) architecture [20] to multiple robots. Sensor failures are diagnosed by allowing the robots to share knowledge of the state of their sensors and task execution via communication. Another approach [21] addresses sensor failures in probabilistic sensor fusion. A p-norm opinion pool method is introduced to detect and exclude faulty measurements. The main drawback of [19,21] is that they only address sensing failures.

Xingyan and Parker [22] have proposed a fault detection approach called SAFDetection ("Sensor Analysis for Fault Detection"). Data clustering and state transition diagram techniques are used in the fault detection process. During training, states and transitions are determined from normal sensor data. Faults are detected in a classification stage where online sensor data are compared with the state transition model. Failures in sensing and actuation hardware can be detected. A tightly coupled box pushing task is used to demonstrate the approach. Xingyan and Parker [23] present a distributed version of the SAFDetection approach to improve scalability and reliability. The SAFDetection approach is well suited for behaviour based systems since the states roughly correspond to behaviours.

A complex artificial immune system is employed for fault tolerant cooperative multi-robot systems in [24]. The artificial immune system mimics the human immune system. Robots are modelled as antibodies, tasks are represented as antigens, and task completion is similar to antigen elimination. The system is fully distributed and capable of accounting for partial and full failures in the robots. However, it relies on detailed information which may not be practical for a reduced human user input approach.

Coalition formation approaches [13,14] have also investigated fault tolerance. In the CoMutaR (Coalition formation based on Multi-tasking Robots) framework [13], sensor failures are addressed by forming new configurations to compensate for missing information. The IQ-ASyMTRe (Information Quality Automated Synthesis of Multi-robot Task solutions through software Reconfiguration) approach [14] also monitors coalitions during task execution. Faults occur when sensor constraints become unsatisfied and the robots need to determine alternative coalitions. CoMutaR and IQ-ASyMTRe are behaviour based methods that address sensing faults only.

Fault tolerance has also been investigated in swarm robotic systems [25,26]. Swarm systems can be implicitly fault tolerant due to redundancy. However, explicit fault tolerance (via explicit detection of failures) can also be beneficial. Failure of wheels in wheeled mobile robots is addressed in [26]. Partial failure, complete failure, and gradual failure are the three modes evaluated. The synchronised flashing behaviour of certain species of fireflies is the motivation for fault tolerance in [25]. Experiments are performed with swarm-bot platforms equipped with rings of LEDs. Periodic flashing of the LEDs functions as a heartbeat mechanism. Failures can be detected when a robot stops flashing its LEDs. The techniques employed in [25,26] are designed for homogeneous systems and may not extend to hierarchical heterogeneous systems.

None of the reviewed fault tolerance methods monitor the four broad categories of robot hardware resources (processing, communication, sensing and actuation) explicitly. Such an approach is required if tasks are specified to robots in terms of these resource categories [5,11]. It is envisioned that this will enable the detection and correction of various types of hardware failures and failures due to poor interaction with the environment (possibly due to incorrect task–robot matching).

3. System overview

Fig. 1 shows an overview of the task allocation and coordination mechanism for the hierarchical heterogeneous multi-robot