Mission design of teams of mobile robots

Matteo Tonelli

Gran Sasso Science Institute
Viale Crispi, 7 L'Aquila

Introduction.

- Autonomous robot are a new powerful instruments that can be involved in several tasks regarding different areas:
  - monitoring.
  - rescue missions.
  - transportations.
  - operations in hazardous environment.
- Robots have limited resources and limited energy autonomy.
- Multiple robot system can be a way to deal whit limited resources.
- Cooperation between robots enlarge the field of application of robots:
  - multiple robot are required for a task.
  - different capabilities of robot are required.

Introduction

Cooperation between robots force system designers to deal with different issues:

- Communication between robots.
- Spatial concurrency.
- Task concurrency.
- Synchronization between robots.
- Fault tolerance.
- Scalability.

The approach proposed to ease complexity is to provide framework that allows designer to define missions in an high level way and manage automatically communication, concurrency, drone failures, navigational error, and the possibility of drones being added or removed.

Characteristic of mission definition languages

The criteria that we should take into account in evaluating a mission definition language are the following:

- Expressiveness.
- Level of abstraction.
- Overhead.
- Scalability and fault tolerance.

Swam programming

Swarm programming is one of the paradigm use to deal with complexity:

- All robots execute a single set of basic rules and operate only on their own local state.
- Complexity are eased by exploiting the simple nature of each robot.
- On the other hand the expressiveness of the languages decrease significantly.
- We will show two example of swarm programming based languages:
  - KARMA.
  - PROTO.

KARMA

KARMA is a swarm based languages tough to define mission for micro-aerial vehicle.

- The system architecture is based on the hive-drone model.
- There are no direct communication between robots, and the robots report the collected data just when they are in the hive.
- The environment is modelled as a topological space and it is divided in different connected region.
- The programmer defines an application as a set of simple robot behaviours along with two functions for each behavior:
  - Activation predicate.
  - Progress function.

PROTO

PROTO is a functional language that provides a behavior programming using an abstraction of the network as a continuous-space-filling device.

- The state of a program is seen as a function defined on the many-fold $M \times T$ where each point describes the state of a device $m \in M$ at time $t \in T$.
- Each device is associated with a neighborhood of nearby devices and knows the state of every device in its neighborhood.
- The primitive offered by PROTO can be divided in four main families:
  - point wise.
  - Restriction.
  - State.
  - Neighbourhood.

VOLTRON

VOLTRON is a programming model that try to express collaborative sensing tasks without exposing the complexity of managing multiple robots.

- The central concept is the *abstract drones* that represents an abstract device that programmers can use to task the robot without individual addressing.
- The most important construct that VOLTRON offer to manage the mobility of robot through the space is *foreachlocation* loop that allows the programmer to specifies the action that has to be performed in each location specified by the user.
- To enable concurrency VOLTRON adopt a system of futures and promises for spatial variables.
- VOLTRON deals time-constraint requirements by giving programmers ways to express time assertions.

A MDE approach

In this approach is proposed a family of domain specific languages that supports the specification of missions and their actual execution. The family are composed of the following languages:

- The Monitoring Mission Language, that is a language especially conceived for domain experts and it is composed by two layer
  - the mission layer that enables the specification of civilian missions.
  - the context layer that contain the definition of the context in which the robot mission will be realized.
- The Robot Language, that describe the type and configuration of the robots that will be in charge of realizing the specified mission.
- The Behavioural Language, that specification of the atomic movements and actions of the robots that have being considered in the mission.

All the model of the family can be combined with transformation in order to produce an actual deployable system.

A Big Actor approach

The concept of BigActors is an extention of the concept of Actor.

- Actor is a model for distributed concurrent computing entities.
  - it communicates using asynchronous message passing.
  - As a response to a message an actor a may compute and change its own local state or send a message to another actor
- The concept of Big Actor use bigraph in order to model dynamically the structure of the world. A bigraph is a combinations of two graphs:
  - a placing graph for modelling nested locality of components;
  - a linking graph for modelling connectivity between components.
- The changeble structure of the world is modelled as set of Bigraph Reaction Rules.
- A bigActor is hosted by a node and it is able to perform the regular actor commands and also to observe the structure of the world, request control actions to change the structure, and migrate from one host to another host.

Thank you for the attention.

Any questions?