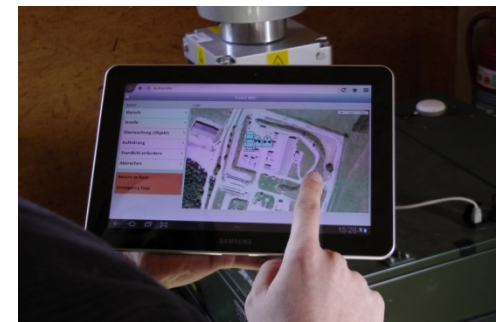
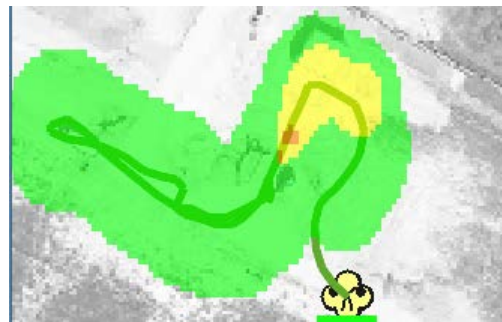
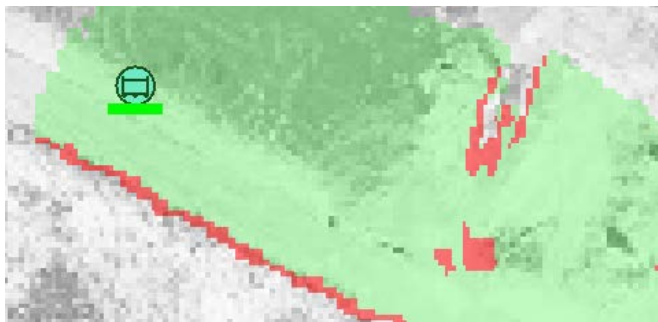


Commanding Heterogeneous Multi-Robot Teams

Thomas Remmersmann – ICCRTS 2014



Motivation

There are many operations in which a multi-robot system (MRS) can be deployed to support the human forces, e.g., for reconnaissance tasks.

Controlling a MRS in operations, however, is a complex and demanding task, especially if the MRS in question has to be controlled by a **single operator** in order to free her fellow soldiers for other tasks and duties.

The operator can be disburdened by giving the robots some **autonomy**.

Motivation

robot autonomy

give orders (task assignments) on a more abstract level;
let the robots handle details themselves.

However, this raises the question as to how those tasks assignments can be defined, formulated and exchanged.

Our approach: express orders (as well as the reports the robots send back to the controller) in **Battle Management Language**.

Battle Management Language

BML has been developed within NATO MSG-048 and NATO MSG-085 (and is discussed by the SISO in order to provide a SISO standard).

BML normally is used to command simulated units in simulation systems in order to improve training, after action analysis, and decision support.

The BML for “C2 system – simulation system”-interaction has been adjusted for our purposes, namely commanding multi robot systems, without changes to the core syntax of the language.

Why Battle Management Language for Robots?

Robots and simulation systems are both systems.

- Both need to “understand” the given commands.

Orders in BML are high-abstraction level orders.

- That’s the way humans give commands.
- They include all the information needed to be executed by humans.
- Long Term Target: Give Robots the same ability.

Additional benefit: connect Robot Systems to existing C2 Systems.

ROS

Robot Operating System

Middleware for R&D projects
Simplifies development

- Defined interfaces
- Interchangeable module
- big, active community

ROS as a middleware

- Simplifies communication
- analyze/monitoring tools
- centrally structured and controlled



The Multi-Robot System Platforms



Longcross Chain

- Weight: ~450kg
- 20 km/h
- 200 kg Payload



RUAG „Garm“ Chain

- Weight: ~500kg
- 20 km/h
- 200 kg Payload

The Multi-Robot System

Payload



- Payload „Autonomous Driving“
- 3D Laser Scanner
 - Xsens positioning (GPS, compass, acceleration sensors)



- Payload „CBRNE“
- Weather station
 - CBRNE-Sensors
 - Xsens positioning (GPS, compass, acceleration sensors)

C2LG GUI

The Graphical User Interface

The screenshot displays the C2LG GUI interface, which is divided into several functional areas:

- Ground Task Panel (Left):** Contains an 'Action' section with a task dropdown set to 'roadrecce' and a grid of icons for various actions. Below this is an 'Emergency Macros' section with a 'STOP' button and a 'Base' button. The 'Units' section shows 'Tasker: schwarm' and 'Taskee: longcross'. The 'Info' section includes 'at-where' (at, poly_139473183077) and 'end-when' (ENDAFT, 3/13/2014, 6:30:20 PM).
- MapView (Center):** A central window showing an aerial map with a large green polygon overlaid on a field. A scale of 1:1.483 is visible. The map includes navigation and tool icons.
- Header and Metadata (Right):** A 'Header' section with fields for 'Sender: schwarm', 'Addressee:', 'MessageID', 'Time: 20140313182803.611', and 'Security:'. Below it are 'Incoming orders' and 'Outgoing orders' sections, each with a table structure for 'OrderID', 'Date', and 'Status'.
- Console (Bottom):** A 'Std Console' window displaying log messages such as 'UnitModel.setParentNode() name = garm node = garm parent = schwarm' and 'BMLComboTextButton.verifyInput(), textName = at-where, matches = true'.

C2LG GUI

The Graphical User Interface

The screenshot displays the C2LG GUI interface, which is divided into three main sections:

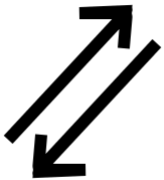
- Task Generation (Left Panel, Orange):** This panel contains a 'Ground Task' section with an 'Action' dropdown set to 'roadrecce'. Below this are several icons representing different task actions. There is also an 'Emergency Macros' section with a 'STOP' button and a 'Base' indicator. The 'Units' section shows 'Tasker: schwarm' and 'Taskee: longcross'. The 'Info' section includes fields for 'at-where' (set to 'at' and 'poly_139473183077'), 'Add CF', and 'end-when' (set to 'ENDAFT', '3/13/2014', and '6:30:20 PM').
- Map Visualization (Center Panel, Green):** This panel shows a topographic map with a green polygon overlaid on it. The text 'map visualization with additional input options for task generation' is overlaid on the map. The map includes a toolbar with various navigation and tool icons, a scale of 1:1.483, and a 'TOC' button.
- Monitoring (Right Panel, Blue):** This panel displays a 'Header' section with fields for 'Sender: schwarm', 'Addressee:', 'MessageID', 'Time: 20140313182803.611', and 'Security:'. Below this are two tables: 'Incoming orders' and 'Outgoing orders'. The 'Incoming orders' table has columns for 'OrderID', 'Date', and 'Status'. The 'Outgoing orders' table also has columns for 'OrderID', 'Date', and 'Status'. There are also buttons for 'Export' and 'TaskID' and 'Status' columns.

Tasks (1)



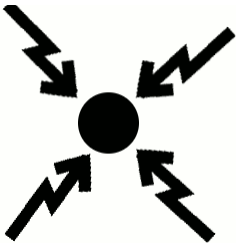
move

Ground vehicle move to position



patrol

A route is secured by the fleet.



observe

A target object is observed
by the MRS.

Tasks (2)



reconnaissance

The MRS is reconnoitering the target area.



imagery intelligence gathering

A picture is taken and reported back.



disengage

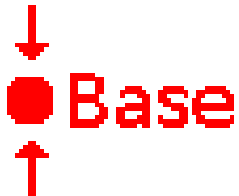
Cancel a task.

Emergency Marcos



Emergency Stop

Cancel all current tasks.

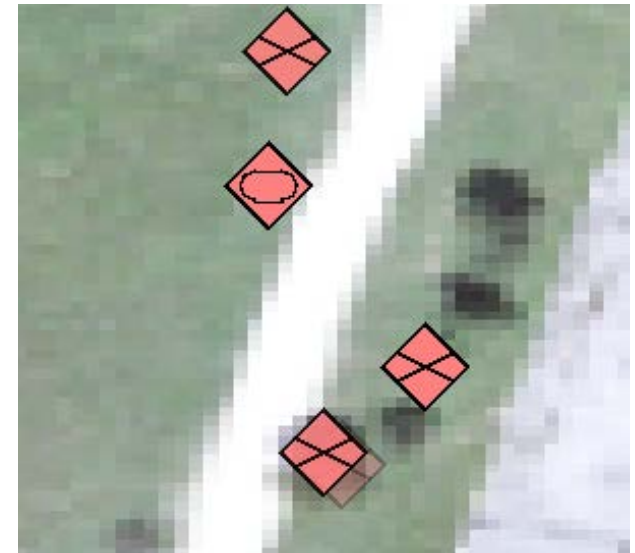
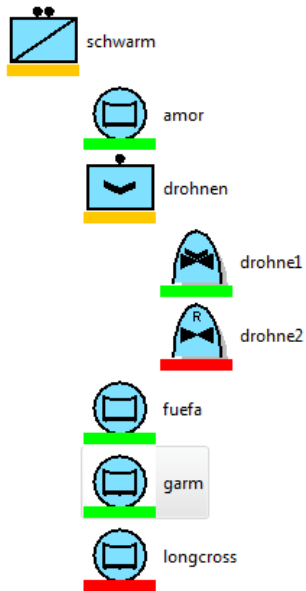


Emergency Return to Base

All robots return to base.

Reports in BML

- Reports are also expressed on “high-level”.
- Aggregate data to produce high-level information.
 - Examples: Robot status, Red-Force Tracking



Reports

Capabilities

The screenshot displays a software interface with a central report table. The table is titled 'NSN Name opCount reqCount' and lists the following data:

NSN Name	opCount	reqCount
Camera	1.0	0.0
GPS	1.0	0.0
Opstate	1.0	0.0
Taskstate	1.0	0.0
Wetter	1.0	0.0
LCD3.3	1.0	0.0

The interface also includes a 'Properties' panel on the left with fields for 'SymbolID', 'Force', and 'Op. Status'. A 'MapView' window is visible in the background, and a 'Unit Tree' on the right lists units like 'schwarm', 'abcrobot', 'garm', and 'longcross'. The Windows taskbar at the bottom shows the system time as 10:33 AM on 4/23/2014.

Planning

Enter High Level Task

The screenshot displays a software interface for mission planning, divided into several panels:

- Ground Task Panel:** Contains an "Action" section with a task dropdown set to "roadrecce" and a grid of icons for navigation and emergency macros. The "Units" section shows "Tasker: schwarm" and "Taskee: longcross". The "Info" section includes "at-where" (at, poly_139473183077) and "end-when" (3/13/2014, 6:30:20 PM).
- MapView Panel:** Shows an aerial map with a large green polygon representing a task area. A scale of 1:1.483 is visible.
- Header Panel:** Displays mission metadata: Sender: schwarm, Addressee: (empty), MessageID: (empty), Time: 20140313182803.611, Security: (empty).
- Incoming orders Panel:** Features an "Import" button and a table with columns "OrderID", "Date", and "Status".
- Outgoing orders Panel:** Features an "Export" button and a table with columns "OrderID", "Date", and "Status".
- Console Panel:** Shows a "Std Console" window with the following log output:

```
UnitModel.setParentNode() name = garm node = garm parent = schwarm
StatusCombo.setText(), text =
BMLComboTextButton.verifyInput(), textName = at-where, matches = true
```

Planning

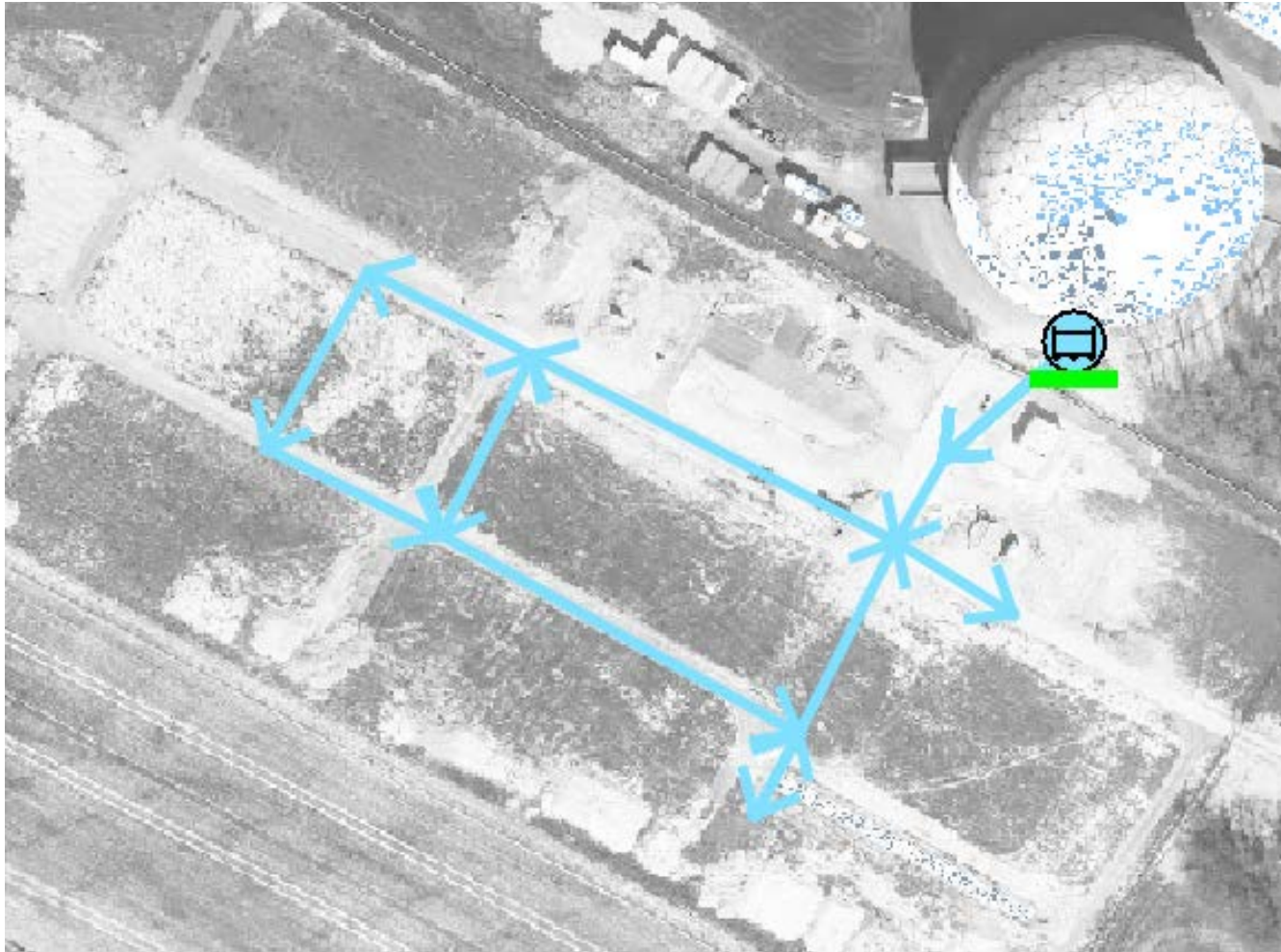
Automatic Process

- relies on descriptions how to get from high level tasks to low level (executable) tasks
- planning system can use known information like
 - previously measured data from MRS like occupancy grids
 - known road network
- planning creates low level tasks, assigns them to respective robots and sorts them, chronologically.



Planning

Show Plan to User via BML



Reports

Sensor Data Interpolation

Visualization of measurements from CBRNE robot

green: harmless

yellow: alarming

orange: hazardous to health

new BML report “Measurement”

Phenomenon

Reporter Identification

Sensor Identification

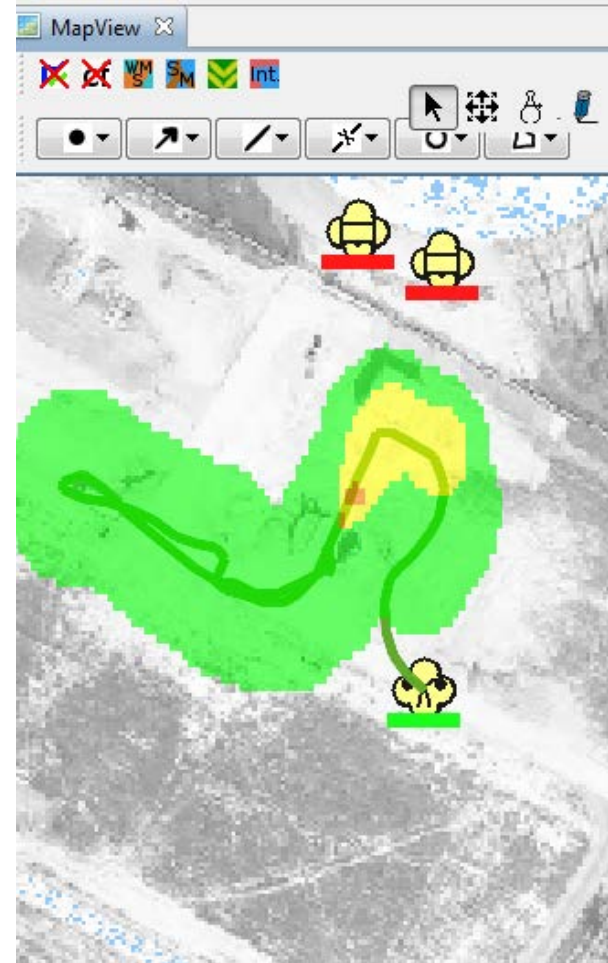
Measured Value

AtWhere

When

Certainty

Label



Report Pictures

Pictures are sent via the
Sensor Data Return Channel

- on demand
- automatically



Demonstration – STARO 2014 CBRNE RECCE
