New Paradigm of Context based Programming-Learning of Intelligent Agents

Andrey V. Gavrilov, PhD, Assoc. Prof.
Department of Production Automation in Machine Building,
Novosibirsk State Technical University,
Email: andr_gavrilov@yahoo.com
Outlines

- Approaches and tendencies in programming of robots and other intelligent agents
- The idea of usage of context in programming of robot
- Requirements to basic language for programming based on context
- CBLR (Context Based Language for Robots)
- Architecture of programming-learning based control system
Existing approaches for programming of robots

- Algorithmic programming (traditional)
  - E.g., languages AML, AL

- Declarative programming
  - Based on description of model of world and tasks

- Visual programming
  - E.g., LEGO Mind Storm NXT, MS Robotics Developer Studio

- Probabilistic programming (for mobile robots)
  - Based on programming of behavior in unknown dynamic environment
Environments for programming

- Robot programming language (traditional)
- Off-line programming environment (e.g., Igrip, CimStation, RobCad, MS Robotics Developer Studio)
  - Based on simulation of robot and using simulation language instead of programming language
- Task-level programming environment (e.g., Ralph, Stanford Smart Robotic Workcell, Aramis, XPROB)
  - Based on description of goals without defining every movement in details:
    - Task specification
    - World model
    - Robot program synthesis
MS Robotics Developer Studio
Ralph structure

- Task statement
- CAD WM
- Sensor data
- Dynamic world database
- Parser & interpreter
- Form & function reasoning
- Planners
- Spatial relationship module
- Servo commands
Features of context based programming

- We suppose that this is kind of task-level programming environment
- Operators are very simple with 1 or 0 parameters to set of value of corresponding context variable
- Delayed action. i.e. any action can be started just when all context variables needed for it have got value
- These features are similar to natural language (NL) and planning of human behavior as execution of commands received from NL based dialog
- This approach is combination of algorithms and events based programming
Architecture of software for transport robot

Knowledge Base with result of learning to understand of NL
Routines for robot
Subsystem for programming
Subsystem for testing
Operating system of robot
Programmer-technologist
Plant

Was developing for automated manufacturing of capacitors

A.V. Gavrilov, Novosibirsk State Technical University, Russia
General paradigm of learning/programming of intelligent equipment. Components

Intelligent equipment (e.g. Robot)

- Actions
- Sensors
- Interpreter of CBLR
- Library of internal functions
- Knowledge base with context variables
- Learning and programming
Levels of languages

- Natural Language
- CBLR
- Internal functions and embedded AI and CI algorithms
- Basic low level language (C or Pascal or Java)
- Plant
  - Sensors
  - Actuators

Intelligent equipment (intelligent agent)

A.V. Gavrilov, Novosibirsk State Technical University, Russia
Fragment of knowledge base with association between collocation ("go", "table") and set of two context variables (#go, #object)
An example of translation of sentence in natural language to CBLR

Go to table and take an apple.

- Go
- Table
- Take
- Apple

#go
#X=100; #Y=200;
#Pick
#Apple
Architecture of programming-learning based control system

- Recognition of images
- Current context
- Concepts
- Recognition (understanding) of natural language
- Speech or text
- Sequence of actions
- Actuators

Sensors

Associative memory

Releaser

Description

Behaviors
# Examples of context primitives

<table>
<thead>
<tr>
<th>Name of context primitive</th>
<th>Possible value</th>
<th>How this parameter influence on execution of motion primitive</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object</td>
<td>Name of object</td>
<td>May be used in action “say”</td>
</tr>
<tr>
<td>Internal state</td>
<td>Good, Bad, Normal</td>
<td>May cause motion to or from Object</td>
</tr>
<tr>
<td>Direction</td>
<td>Left, Right, Forward, Back</td>
<td>May cause corresponding turn depending on Internal state</td>
</tr>
<tr>
<td>Person</td>
<td>Name of person</td>
<td>May be used in action “say”</td>
</tr>
<tr>
<td>Obstacle distance</td>
<td>Far, Middle, Close</td>
<td>May be used in “act”</td>
</tr>
<tr>
<td>Obstacle type</td>
<td>Static, Dynamic</td>
<td>May be used in “act”</td>
</tr>
<tr>
<td>Speed</td>
<td>Low, Normal, High</td>
<td>May be used in “act”</td>
</tr>
</tbody>
</table>
Screenshot of software for simulation of mobile robot with dialog in natural language and CBLR (on going)
Conclusions

- In this report an approach for programming-learning of robots and other intelligent agents was proposed based on dialog by natural language and widely using context.
- Advantages of this approach is opportunity to describe tasks, behavior, movements, world in natural language as learning of robot/equipment.
- Result of learning-programming is description of behaviors releasing by recognized concepts of context.
- Now this approach is implementing for mobile robots oriented for navigation tasks in our simulation system MRS.
Thanks!

Andr_gavrilov@yahoo.com