Conditional Planning in HRI

Advantages
- Does not require complete knowledge about the initial state at planning time (can model situations where some user needs are not known)
- Allows for minimal execution of sensing (reduces wrong behaviors due to incorrect perception)

Limitations
- Not too many planners available
- Writing planning domains is still difficult
- Loops (e.g., while conditions) are not

Three planners tested
- HATP
- ROSPlan with Contingent-FF
- MDP solver

Execution framework
- Petri Net Plans
- Execution rules
Conditional Planning in HRI

Petri Net Plans

- Plan representation formalism
- Complex robot/multi-robot/human-robot plans
- Implicit representation of uncertainty
- Explicit representation of sensing actions
- Execution monitor
  - Interrupt
  - Execution conditions and recovery procedures

[Ziparo et al., 2011]
Petri Net Plans

- Ordinary and sensing actions
- Conditions and loops
- Interrupts
- Parallel execution (fork and join operators)
- Multi-robot support

PNPGen generates PNP from the output of several planners (MDP solver, ROSPlan, HATP, …)

PNP-ROS uses ROS actionlib to run plans including ROS actions

Petri Net Plans

Ordinary and sensing actions

L. Iocchi - Human-Robot Interaction
Conditional structures

While loops
Petri Net Plans

Interrupts

Parallel execution
Petri Net Plans

Multi-agent synchronization

Manual writing of PNP through graphical editor NOT SCALABLE
Petri Net Plans

Automatic generation of PNP from planner outputs
- HATP + Sensing
- ROSPlan + Conditional Planner
- MDP solver

In particular, automatic generation of conditional plans for HRI

HATP + Sensing

- On-line negotiations in human-robot collaborative tasks
- On-line sensing and conditional plans
- Execution flow of plan depends on on-line sensing
- Execution variables (determined at execution time)

https://bitbucket.org/EugSeb/hatp-pnpplanner

[Sebastiani et al., 2017]
Use Case 1

Robot helps to bring objects
(15 actions, 44 places, 41 transitions, 86 edges)
Use Case 2

Collaborative navigation

Robot and human have to reach a target location with small passages, 3 doors to be opened, 2 elevators to be operated

(410 actions, 1128 places, 1098 transitions, 2254 edges)

ROSPlan+Sensing

ROSPlan generates a conditional plan

PNP executes the conditional plan
Advantages

- No assumption (about HRI) made at planning time
- Reduced need of state estimation
- Local recovery procedures

https://sites.google.com/site/rosplantopnp/

[Sanelli et al., 2017]
**MDP+Sensing**

- Compact representation of an MDP through PRU+ language
- MDP solver
- Generation of PNP from policy (with minimal sensing)

[Iocchi et al., 2016]

**Main features**

- Minimal sensing (sensing actions on relevant state variables)
COACHES demos

Demo 1. Advertising in many shops

Demo 2. Approaching and answering user requests

Manual writing of cond. plans

- Faster, but not scalable

Example

display_init; waitfor_screentouched; display_text_welcome;
display_text_name; display_text_intro; display_text_info;
ask_whichentertainment; waitfor_HRIreceived; < question ? choose_category;
waitfor_category.@Cat; choose.@Cat; waitfor_element.@Elem; LABEL1;
asking_question.@Cat.@Elem; waitfor_HRIreceived; < answer_right ?
display_image_answer_right; display_text_answer_right : answer_wrong ?
display_image_answer_wrong; display_text_answer_wrong; GOTO LABEL1
>; : quiz ? choose_quiz; waitfor_quiz.@Z; LABEL2; ask_quiz.@Z;
waitfor_HRIreceived; < answer_right ? display_image_answer_right;
display_text_answer_right : answer_wrong ? display_image_answer_wrong;
display_text_answer_wrong; GOTO LABEL2 >; : joke ? choose_joke;
waitfor_joke.@J; display_text_joke.@J >; display_image_answer_right;
display_text_goodbye; restart
Execution rules and robustification

Execution rules

Adding to the conditional plan
- interrupt (special conditions that determine interruption of an action)
- recovery paths (how to recovery from an interrupt)
- social norms
- parallel execution

Main feature
- Execution variables are generally different from the ones in the planning domain (thus not affecting complexity of planning)
Examples

if personhere and closetotarget during goto do skip_action
if personhere and not closetotarget during goto do say_hello; waitfor_not_personhere; restart_action
if lowbattery during * do recharge; fail_plan
after receivedhelp do say_thanks
after endinteraction do say_goodbye

when say do display
Robot Office Assistant

Run 1 - without Execution Rules

References

