KeJia Project: Towards Integrated Intelligence for Service Robots

Xiaoping Chen, Guoqiang Jin, Jianmin Ji, Feng Wang and Jiongkun Xie

Multi-Agent Systems Lab., Department of Computer Science and Technology,
University of Science and Technology of China, HeFei, 230027, China
xpchen@ustc.edu.cn
http://wrighteagle.org/en/robocup/atHome

Abstract. This paper reports some progress on the project KeJia, a long-term effort aiming at integrated intelligence for service robots. We focus in this paper on the high-level cognitive functions, including human-robot dialogue understanding, task planning, and automatic transformation of the knowledge and information extracted from human-robot dialogues and other sources in verbal languages into the planner. We have been testing these techniques and the integrated system in RoboCup@home standard tests and other case-studies involving general purpose service with incomplete or erroneous information, acquiring and reasoning with causal knowledge, and learning to operate a microwave oven through reading the manual.

1 Introduction

Researchers from Artificial Intelligence (AI), Robotics and related areas have shown increasing interest in developing intelligent service robots [1, 2, 5, 7, 14, 17] A service robot is generally regarded as a robot servant providing services for untrained and non-technical users in ordinary environments such as home, office, and hospital.

The motivation behind the work reported here is to try to develop intelligent service robots that meet the following three challenging requirements. Firstly, an intelligent service robot should be able to communicate with humans naturally [1, 10, 5, 8]; in particular, it should be able to understand its users' service requests and other messages expressed in some verbal languages. Secondly, an intelligent service robot should possess some degree of autonomy; in particular, it should be able to carry out task planning autonomously. Thirdly, an intelligent service robot should be able to learn from its experience and thus reach higher performance; in particular, we hope the robot can acquire general knowledge from human users through spoken dialogue and other sources such as the web. To our best knowledge, there is very little work on this particular requirement, while there are lots on robot learning, in which a human teaches a robot how to perform a specific task through a combination of spoken commands, observation and imitation of the human's performing that task [16].

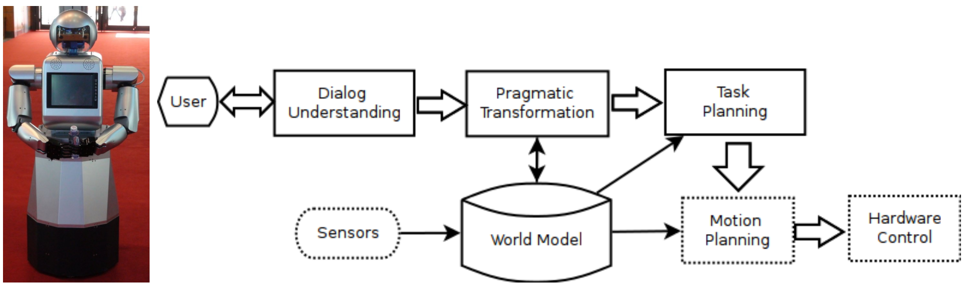
In a long-term project aiming at the three requirements [5, 6], some general-purpose mechanisms for processing limited segments of natural languages (LSNLs), task planning, and declarative knowledge acquisition are developed and implemented on a real robot KeJia. We have tested these techniques and the whole system in RoboCup@home league competitions in past two years [4, 3] as well as other case-studies. In this paper, which serves as the team description paper of WrightEalge for RoboCup@home 2011, we concern ourselves with some progress during the last year.

Section 2 gives an overview of KeJia system. Section 3 describes a key module of KeJia system, Pragmatic Transformation. Section 4 presents hierarchical task planning. Section 5 describes briefly the low-level functions and Section 6 reports some experiments on KeJia. Conclusions are given in Section 7.

2 The Realization of Real Robot KeJia

We have made a new robot, KeJia-2, as shown in figure 1a. Its hardware components include a laser range finder, a stereo camera, and two arms for manipulating portable items, all installed on an omnidirectional wheeled base. The computational resources consist of two on-board 4-core PCs. Neither additional computational resources off-board nor remote control is needed for the robot when it performs its tasks.

The flowchart of KeJia system is shown in Figure 1b. The robot is driven by input from human-robot dialogue. The spoken dialogue between robot KeJia and its users is restricted to some limited segments of natural languages (LSNLs). A specific LSNL is defined with a fixed vocabulary and a simplified syntax. Service queries, descriptions about the states of the environment, knowledge of the world, instructions about new tasks and so on can be expressed in these LSNLs.



(a) KeJia-2 (b) The flowchart of KeJia's high-level functions

Fig. 1: Hardware and Software of KeJia

The texts drawn by standard speech recognition software from the spoken dialogue is processed syntactically with the Stanford parser [12], and then semantically through a lazy semantic interpreter we developed in KeJia project [5]. The results of the semantic analysis are represented in a form similar to the Discourse Representation Structure (DRS) [11], with an extended semantics. The information in this internal representation is transformed by the pragmatic transformation module (see Section 3 for more details) into the Task Planning module, which is based on Answer Set Programming (ASP) [9], where the information and knowledge are represented as an ASP program, and some ASP solver is employed to generate a course of actions for the user's task.

A high-level plan generated by the Task Planner is fed into the Motion Planner. Each action is designed as a primitive for KeJia's task planning and can be realized by KeJia's motion planner and then executed by the Robot Controller autonomously. The execution causes some changes of the environment and the state of the robot itself. The World Model is updated accordingly with the information perceived by the sensors.

The Motion Planner deals with a repertoire of (low-level) routines and predefined parameters. For each low-level function of the robot, such as object recognition and manipulation, there is a routine, which involve uncertainties that could be best modeled with quantitative mathematical methods.

3 The Pragmatic Transformation

The Pragmatic Transformation is one of KeJia's key components, with which there is very little work in literature. Its input is a set of LSNL-sentences in some internal representation. Its output is an ASP program which contains all the information for solving the user task. The pragmatic transformation is completed by executing a set of transformation rules, which are generally built up on top of another kind of more elementary rules, called interpretation rules.

The interpretation is not a linguistic issue, but the symbolic grounding problem [17], i.e., how to link the abstract concepts (expressed with the words from LSNLs in KeJia system) to the perception and actuation of the robot.

1. Interpretation Rules. Here we only describe three main sorts of primitive interpretation rules informally. The first sort is for nouns, pronouns and adjectives in LSNLs, which are linked directly to the World Model and thus mapped into the low-level abstraction of the robots perceptual data. In particular, a noun or pronoun is interpreted as an object, a set of objects, or an attribute of the World Model, while an adjective an attribute of the World Model.

The second sort of primitive interpretation rules is for verbs in LSNLs. A verb is eventually mapped into a course of low-level actions of the robot On the other hand, most verbs should not be mapped directly into the robots routines; otherwise, there would be no task and/or motion planning for the realization of the actions that these verbs refer to. To support the specification of verbs mapping, we introduced a substrate symbolic language, which consists of primitive actionsand other built-in identifiers, which are defined and realized in terms of the robots routines and parameters.

The third sort of primitive interpretation rules is for words and linguistic constructors corresponding to logical operators. In KeJia Project, we have considered two words corresponding to two most important logical operators, "if" and "not" [5]. There are three cases where "not" appears in current LSNL-sentences. (i) "not" is used to form a negative, imperative sentence, such as "do not open the door". The whole sentence expresses that some action is forbidden, which can be translated naturally into an ASP constraint. (ii) "not" modifies the main verb of a sentence or clause. These sentences or clauses should be handled similarly to a negative, imperative sentence as above. (iii) "not" modifies "anything", representing "nothing". The sentence or clause should be translated into an ASP rule too. In all the cases, word "not" is translated naturally or approximately into the negation-as-failure operator, not.

2. Transformation rules. For sake of simplicity, we omit the first-order format of DRS here and describe the transformation rules as mappings from LSNL-sentences into ASP-rules. KeJia's semantic analyzer classifies the LSNL-sentences into three types, for each of which there is a set of corresponding transformation rules.

(a) LSNL-sentences that just provide information about the environment will be transformed into ASP facts and/or rules. For instance, "The book is on the table." is converted to the following ASP rules:

$$holds(samelocation(X,Y),0) \leftarrow book(X), table(Y).$$

where "samelocation" is a built-in identifier of the substrate symbolic language, which is interpreted by pre-defined parameters, such as some gridding of the environment. With this rule, when needed, the robot will search the book at the same location of the table according to the parameters, by executing the corresponding routine.

(b) A LSNL-sentence representing a simple task will be transformed into an ASP goal. For instance, "Give Jim a red bottle" is a simple task and will be transformed as following ASP-rules:

$$\begin{aligned} goal \leftarrow & holds(samelocation(X,Y), lasttime), holds(handempty, lasttime), \\ & Jim(X), red(Y), bottle(Y). \\ \leftarrow & not\, goal. \end{aligned}$$

(c) Causal knowledge are also transformed into ASP rules. Consider the sentence: "the object will fall, if the object is on the sticking-out end of the board and there is nothing on the other end of the board." This sentence expresses a piece of knowledge regarding a special form of notion of balance. Based on relevant interpretation rules, this is translated to the following ASP rule:

$$\begin{aligned} holds(falling(X),T) \leftarrow & holds(on(X,Y),T), sticking\_out(Y), endof(Y,Z), \\ & board(Z), endof(U,Z), not\, holds(on(V,U),T). \end{aligned}$$

where sticking_out, endof and board are built-in identifiers and can be handled by the Motion Planner and the robot's perceptual system.

According to the interpretation and transformation rules, other domain knowledge can be converted and added into ASP program similarly.

4 Hierarchical Task Planning

In KeJia's task planner, a planning problem is described as an ASP program, and an ASP solver is called to get its answer sets, each corresponding to a high-level plan of the problem. For small problems that have a plan with less than 20 steps, this works well. But this is not the case when the problem is large, since the current ASP solvers are not efficient enough. In domestic domains, a typical task such as "clean the house" may contain much more steps. We have tested a 47 steps problem, and it took 25 hours to get a single solution.

Along with the development of ASP solvers, the ASP planning technique is hopeful to be able to handle larger and larger problems in the future. Meanwhile, there are other opportunities of speeding up solutions with the current ASP solvers. In planning, the longer a plan is, the more time will be spent on one step forwarding. So it is not surprising that a 20 steps plan takes twice the time as a 19 steps plan for the same problem. Thus if we can shorten the plan length, the time for solving the problem can be greatly saved. One of the promising techniques is to use macro-actions in the planning.

A macro-action represents a sequence of primitive actions of the domain. In planning procedure a macro-action acts just as a primitive action, added to the original domains. When the adapted problem is solved, a plan including macro-actions is generated. Then all the macro-actions are refined to primitive actions.

Currently we are using two types of macro-actions in KeJia's system. First one is the "Relevant Object Macros (ROMs)", where a pre-defined sequence of primitive actions is used to accomplish a sub-task or to handle a certain object with multiple primitive actions sequently. The second one consists of those macro-actions learned from small-size problems of the same domain.

Some macro-actions can be refined straightforwardly, that is, replaced by the corresponding primitive action sequences. But the replacement may be difficult or even impossible in some cases. A more general way is to take the refinement of a macro-action as an induced, new planning problem. In the new problem, the initial state is the state before the macro-action's execution, the goal state is the state after its execution, and the actions are all primitive.

With the hierarchical planning method, KeJia completes task planning much more efficiently. For example, for the problem which has a 47 steps optimal plan mentioned above, KeJia got a 48 steps plan in 40 seconds with the method.

5 Low-level Functions

Low-level functions are necessary and crucial in some cases for the development of an intelligent service robot. Here we describe briefly navigation, perception, and manipulation implemented on our robot KeJia.

Navigation A 2D occupancy grid map is learned from laser scans, collected by the robot though a round travel within the rooms aforehand [10]. The map is then manually annotated with the approximate location and/or area of rooms, doors, furniture and other interested objects. And thus a topological map can be automatically generated, which will be used by the global path planner and imported as a part of prior world model. Scanning match and probabilistic techniques are employed for localization, and VFH+ [18] is adopted to avoid a local obstacles while the robot is navigating in the rooms.

Perception We follow the approach proposed in [15] to detect and locate the tabletop objects, such as bottles, cups, appliances. A further effort was taken to make the robot be able to carry out challenging manipulation task e.g. operating household appliance. Take the microwave oven for example, to preciously estimate the 6-DOF pose of the oven's body, buttons and even the opening angle of the oven's door, our current implementation employs a model-based method, which aligns 3D point clouds from stereo vision with the given geometric model of the oven and achieves a repeatable accuracy of less than 1 mm.

Manipulation We simplified the algorithm described in [13] by tracking a set of marks attached to arm mechanism, rather than the articulated point cloud model of the arm, to perform online hand-eye calibration and coordination. The online calibration error of the vision-manipulator system can be less than 5 mm while the arm stops moving, which greatly improves the success ratio of manipulation.

6 Experiments

We have conducted a series of case studies in KeJia project.Some of the case studies that aim to examine the high-level cognitive functions are described briefly below.

General Purpose Service with Incomplete or Erroneous Information: In an experiment[1], the user firstly gave an underspecified instruction to KeJia. KeJia discovered this underspecification and actively requested for more information to find out the "true" intention of the user and then completed the task accordingly. In addition, the user asked KeJia to perform an unachievable task. KeJia found the infeasibility of the task during the execution of her tentative plan and reported the infeasibility to the user, instead of blind execution of user's commands regardless of any potential problems.

Acquiring from Spoken Dialogue and Reasoning with Causal Knowledge: In the experiments, we took a version of KeJia without any build-in knowledge about "balance", "fall", or any other equivalents. KeJia was told in the human-robot spoken dialogue that an object will fall if it is on the sticking-out end of a board and there is nothing on the other end of the board. With the knowledge, KeJia accomplished the task of moving the green can while avoiding anything falling with the following plan: moving the red can and put it on the table first, and then picking up the green can.

[1] The video is at http://wrighteagle.org/en/robocup/atHome/video/WEHome2011Quali.wmv

Learn to Operate a Microwave Oven through Reading the Manual: This demo[2] was repeated dozens of times to the public in the 12th China Hi-tech Fair (Shenzhen, Nov. 16-21, 2010). During the demonstration, the user gave oral commands in Chinese to KeJia, requesting her to prepare breakfast, in particular, to heat up milk and/or bread. In the beginning, KeJia had no complete knowledge about the microwave oven; for instance, she did not know the functions of the buttons on the control panel. While the user suggested her "reading the manual", KeJia immediately browsed and downloaded the manual, read the manual in Chinese and learnt the relevant knowledge in it. With the acquired knowledge, KeJia managed to plan a course of actions and achieve the entire task. The actions she executed included opening the microwave oven door, putting the food into the microwave oven, closing the door, pushing button "Start/Re-heat" to start heating, pushing button "Pause/Cancel" to stop heating, opening the oven and finally bringing the heated food to the user. In this process, KeJia checked her actions of pushing buttons by monitoring the readings on the digital screen, and took remedial actions when necessary.

7 Conclusions

In order to meet the requirements mentioned in Section 1, we are developing and integrating techniques for natural language understanding for limited fragments of English and China, hierarchical task planning, and automatic transformation of the knowledge and information drawn from human-robot dialogue and web pages into some form available by the task planner. We are also developing Low-level functions that are necessary for implementing an intelligent service robot, including navigation, perception, and manipulation. In order to test these techniques and the entire system, we have conducted a series of case studies involving general purpose service with incomplete or erroneous information, acquiring and reasoning with causal knowledge, and learning to operate a microwave oven through reading the manual.

Acknowledgement This work is supported by the National Hi-Tech Project of China under grant 2008AA01Z150, the Natural Science Foundations of China under grant 60745002, and USTC 985 project. Other team members besides the authors are: Kai Chen, Min Cheng, Xiang Ke, Zhiqiang Sui.

References

1. H. Asoh, Y. Motomura, F. Asano, I. Hara, S. Hayamizu, K. Itou, T. Kurita, T. Matsui, N. Vlassis, R. Bunschoten, et al. Jijo-2: An office robot that communicates and learns. *Intelligent Systems, IEEE*, 16(5):46–55, 2005.
2. W. Burgard, A. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun. Experiences with an interactive museum tour-guide robot. *Artificial Intelligence*, 114(1-2):3–55, 1999.

[2] http://www.wrighteagle.org/en/demo/microwaveoven.php

3. X. Chen, J. Ji, J. Jiang, and G. Jin. WrightEagle Team Description for RoboCup@ Home 2009. Technical report, Technical report, Department of Computer Science and Technology, University of Science and Technology of China, 2009.
4. X. Chen, J. Ji, J. Jiang, and G. Jin. Progress of Ke Jia Project. Technical report, Technical report, Department of Computer Science and Technology, University of Science and Technology of China, 2010.
5. X. Chen, J. Ji, J. Jiang, G. Jin, F. Wang, and J. Xie. Developing high-level cognitive functions for service robots. In *Proceedings of the Ninth International Conference on Autonomous Agents and Multiagent Systems (AAMAS-10)*, pages 989–996, 2010.
6. X. Chen, J. Jiang, J. Ji, G. Jin, and F. Wang. Integrating nlp with reasoning about actions for autonomous agents communicating with humans. In *Proceedings of the 2009 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT-09)*, pages 137–140, 2009.
7. A. Ferrein and G. Lakemeyer. Logic-based robot control in highly dynamic domains. *Robotics and Autonomous Systems*, 56(11):980–991, 2008.
8. T. Fong, I. Nourbakhsh, and K. Dautenhahn. A survey of socially interactive robots. *Robotics and autonomous systems*, 42(3-4):143–166, 2003.
9. M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In *ICLP/SLP*, pages 1070–1080, 1988.
10. G. Grisetti, C. Stachniss, and W. Burgard. Improved techniques for grid mapping with rao-blackwellized particle filters. *Robotics, IEEE Transactions on*, 23(1):34–46, 2007.
11. H. Kamp and U. Reyle. From discourse to logic: Introduction to modeltheoretic semantics of natural language, formal logic and discourse representation theory. *Computational Linguistics*, 21(2):265–268.
12. D. Klein and C. Manning. Fast exact inference with a factored model for natural language parsing. *Advances in neural information processing systems*, pages 3–10, 2003.
13. M. Krainin, P. Henry, X. Ren, and D. Fox. Manipulator and object tracking for in hand model acquisition. In *Proc. of the Workshop on Best Practice in 3D Perception and Modeling for Mobile Manipulation at the Int. Conf. on Robotics & Automation (ICRA), Anchorage, Alaska*, 2010.
14. M. Quigley, E. Berger, A. Ng, et al. Stair: Hardware and software architecture. In *AAAI 2007 Robotics Workshop, Vancouver, BC*, 2007.
15. R. Rusu, A. Holzbach, M. Beetz, and G. Bradski. Detecting and segmenting objects for mobile manipulation. In *Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on*, pages 47–54. IEEE, 2010.
16. P. Rybski, K. Yoon, J. Stolarz, and M. Veloso. Interactive robot task training through dialog and demonstration. In *Proceedings of the ACM/IEEE international conference on Human-robot interaction(HRI-07)*, pages 49–56. ACM, 2007.
17. M. Tenorth and M. Beetz. KnowRobknowledge processing for autonomous personal robots. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 4261–4266. IEEE, 2009.
18. I. Ulrich and J. Borenstein. VFH+: Reliable obstacle avoidance for fast mobile robots. In *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, volume 2, pages 1572–1577. IEEE, 2002.