POLITECNICO DI TORINO
Repository ISTITUZIONALE

Complex Task Planning for General-Purpose Service Robots

Original
Complex Task Planning for General-Purpose Service Robots / Contreras, L.A., Savage, J., Ortuno Chanelo, S., Negrete, M., Okada, H.. - In: JOURNAL OF ROBOTICS AND MECHATRONICS. - ISSN 0915-3942. - 37:3(2025), pp. 594-603. [10.20965/jrm.2025.p0594]

Availability:
This version is available at: 11583/3005660 since: 2025-12-05T10:40:26Z

Publisher:
Fuji Technology Press

Published
DOI:10.20965/jrm.2025.p0594

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

26 November 2026

https://doi.org/10.20965/jrm.2025.p0594

Paper:

Complex Task Planning for General-Purpose Service Robots

Luis Contreras*, Jesús Savage, Stephany Ortuño-Chanelo***, Marco Negrete**, and Hiroyuki Okada***

*Advanced Intelligence & Robotics Research Center, Tamagawa University
6-1-1 Tamagawagakuen, Machida, Tokyo 194-8610, Japan
E-mail: contreras.luis@lab.tamagawa.ac.jp
**Signal Processing Department, National Autonomous University of Mexico
Circuito Exterior S/N, Ciudad Universitaria, Coyoacán, Mexico City 04510, Mexico
***Visual and Multimodal Applied Learning Laboratory, Polytechnic University of Turin
Torino 10129, Italy
[Received March 19, 2024; accepted November 12, 2024]

Service robots, in contrast to industrial robots, are devices aimed to operate in the service sector in replacement of or collaboration with human performers—in particular, domestic service robots carry out daily household chores. Recently, their popularity has increased and the range and difficulty of the activities they can perform are reaching amazing performances. There have been proposed several systems that can generalize single tasks such as object manipulation or scene understanding but they still fail on complex task planning, i.e., when and where they can perform such tasks. In this work, we propose the use of expert systems as a core module to understand user commands, infer the task context, request missing information, and plan action where each step of the plan may consist of basic state machines to generalized deep-learning models.

Keywords: domestic robots, general purpose service robots, expert systems, task planning, semantic understanding

1. Introduction

The presence of service robots in daily activities is increasing recently and, in consequence, the familiarity of users and the difficulty of the tasks that they might request are increasing. Personal robots, in contrast to industrial robots, are machines that can sense, make plans, take decisions, and then act towards accomplishing a specific human-inspired task; a service robot is a personal robot aimed to help the service sector and therefore, it has to solve tasks such as security, cleaning, medical, and human support [1].

From developing individual skills to performing complex tasks, several platforms have been proposed recently aimed at solving general-purpose tasks. However, although many platforms and tests have been proposed, they mainly focus on single tasks at a time, like "Take out the garbage," "Storing groceries," "Clean up," "Welcoming visitors," and so on. Furthermore, many of those tests are performed by stationary robots, consisting mainly of an RGB(D) camera and a robotic arm manipulating objects on a surface.

Our current work focuses on action planning by using semantic reasoning to generate a global plan from a given command (spoken or in a command shell) while we integrate standard and data-driven skill models with human-intention detection and active reasoning to generate local plans for low-level behaviours.

2. Related Work

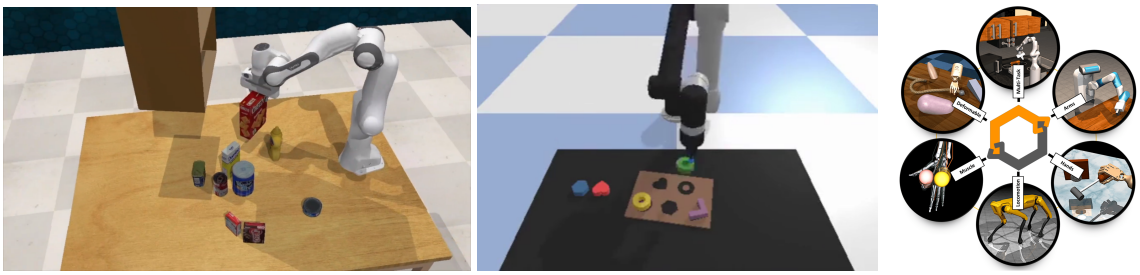
To assess the state of the art on general-purpose service robots, we propose using robot competitions as a benchmark. A popular service robot competition is RoboCup@Home [a], where a robot should perform a series of housekeeping and house-party domestic tasks, act as a waiter in a restaurant, and, as mentioned before, perform a general purpose test where the robot receives randomly generated spoken commands; the task the robot should execute include storing groceries, welcoming guests, carry someone's luggage, and setting a table.

Then, in the European Robotics League (ERL [b]) we have the Consumer Service Robots Challenge [c]—known before as RockIn@Home [d]—where the robot performs some of the duties of a caregiver, including welcoming visitors and catering for an elderly person comfort.

Next, extending the application of service robots to indoor and outdoor environments, the ERL Smart City Robotics Challenge [e] includes tasks in several spaces such as assisting the staff of a coffee shop to take care of their customers, picking products from a storage container and placing them on a designated shelf, and delivering medicines to a person across the city.

Also, appealing to some semantic reasoning, in the Roboethics Competition [f], teams develop a robot that fetches objects in a home from and to several users with different features such as age, rank, relationship, and so on, where participants are asked to solve challenging ethi-

© Fuji Technology Press Ltd. Creative Commons CC BY-ND: This is an Open Access article distributed under the terms of the Creative Commons Attribution-NoDerivatives 4.0 International License (https://creativecommons.org/licenses/by-nd/4.0/)



(a) (b) (c)

Fig. 1. Although several systems and benchmarks able to perform and evaluate robot performances have been developed, recently, these systems seek to skill generalization, where the same system is able to solve several similar tasks without further training. Examples are: (a) RLBench [4], (b) Ravens [5], and (c) RoboHive [6].

cal situations for a robot in scenarios with potentially hazardous objects in the house (including alcoholic drinks, sharp items, and drugs and medicines).

Similarly, in SIGVerse [2], the authors develop a virtual reality (VR) platform for human-robot interaction research that has been applied to service robots interaction in both directions: from and to the human. Tasks include "Interactive cleanup," where the human points to target objects and deposits and "Human navigation," where the robot has to generate friendly and simple natural language expressions to guide people to perform several tasks in a daily environment.

Finally, the main goal of World Robot Summit (WRS) [3] is to evaluate the basic skills of the robot to complete the core functions of a home assistant, such as cleaning a room, delivering items, setting a table, and so on by following a series of metrics and benchmarks that allows teams to consistently measure the performance for a given solution to this challenge.

Recently, RLBench, a test platform for several manipulation applications, has been presented in [4], where the authors use a simulator with a fixed robot arm on a surface with an upper RGBD camera and an eye-in-hand monocular camera, and they feature 100 different tasks providing proprioceptive and visual observations. Similarly, in [5] the authors present Ravens, a simulated benchmark with ten manipulation tasks, mainly focusing on the "transportation" skills, i.e., tasks that execute a sequence of spatial displacements. Furthermore, RoboHive, as in [6], encompasses a diverse range of pre-existing and novel environments organized within, covers multiple domains such as hand manipulation, locomotion, multi-task, multi-agent, muscles, etc., and establishes metrics for evaluating algorithm performance across all environments.

In **Fig. 1** we can observe the different tasks these systems should solve; as we can observe, they are mainly robotic manipulators with one or several cameras performing object manipulation. Although they extend a single model to several tasks, it is a step closer to general-purpose service robots.

More general works towards service robots are Habitat [7, 8], sDSPL [9], and BEHAVIOR [10]. In Habitat, the authors present a simulated home environment where the robot can perform several tasks where the latest version allows for real human interaction with simulated robots; however, no specific rules and regulations for given tasks are provided. In this sense, the authors propose the simulation Domestic Standard Platform League (sDSPL) as a benchmark to evaluate the performance of service robots while executing a task; they focused on standardization and evaluation of service robots while performing a general purpose task to close the gap between robot competitions and research.

Finally, in BEHAVIOR the authors present their "Benchmark for everyday household activities in virtual, interactive, and ecological environments;" they propose 100 indoor chores in realistic environments such as "Assembling gift baskets," "Cleaning bedroom," "Installing a modem," or "Sorting groceries." They provide 500 human demonstrations in virtual reality and propose a number of a set of metrics to measure task progress and efficiency, absolute and relative to human demonstrators. This work is built under iGibson 2.0 [11], a simulation environment that supports object states (such as temperature, wetness level, and cleanliness level) and implements a set of functions that map physical states (like cooked or soaked); moreover, it includes a VR interface to immerse humans in its scenes to collect demonstrations that can be used for imitation learning. For comparison, in **Fig. 2** we can observe these three virtual systems.

3. General-Purpose Tasks for Service Robots

One key challenge towards autonomous service robots has been proposed in the RoboCup@Home competition, namely, the general purpose service robot (GPSR) test, where several general-purpose commands are randomly generated using a [EE]GPSR command generator and grammars publicly available at [g].

In short, in this test, there are three categories according to their difficulty. In category I, the service robots must solve easy tasks involving basic skills such as indoor navigation, grasping known objects, answering questions (from the predefined set of questions), etc. (e.g., "Bring me the apple juice from the counter," "Tell me how many beverages are on the shelf," "Tell me the name of the person at



(a) (b) (c)

Fig. 2. Recently, several virtual platforms for domestic robots have been developed where complex environments and their physical states can be simulated, allowing for general-purpose tasks to be evaluated. These systems include: (a) Habitat 2.0 [7], (b) sDSPL [9], and (c) BEHAVIOR [10].

the door"). In category II, robots must solve tasks with a moderate degree of difficulty. This category involves following a human, indoor navigation in crowded environments, manipulation and recognition of alike objects, finding a calling person (waving or shouting), etc. (e.g., "Tell me how many beverages on the shelf are red," "Count the waiving people in the living room," "Follow Anna at the entrance"). Finally, in category III, the robots must understand challenging tasks dealing with incomplete information, environmental reasoning, feature detection, natural language processing, outdoor navigation, pouring, opening doors, etc. Examples are: "Pour some cereals in the bowl," "Go to the bathroom (the bathroom door is closed)," "Bring me the milk from the microwave (the milk is inside the microwave)."

Moreover, there is also an extended GPSR (the EE-GPSR) test, where the robot has to perform complex actions, given as a single sentence while validating the information provided in the commands through HRI (e.g., by asking for missing information). Some examples are:

- Go to the kitchen counter, take the coke, and bring it to me.
- Bring the chips to Mary on the sofa, tell her the time, and follow her.
- Find a person in the living room, guide them to the kitchen, and follow them.

4. Expert-System Based GPSR

Towards task generalization, CLIPort [12] proposes an end-to-end framework that is capable of solving a variety of language-specified tabletop tasks from packing unseen objects to folding cloths, all without any explicit representations of object poses, instance segmentation, memory, symbolic states, or syntactic structures; this work combines the manipulation skills of Transporter [5] with the image representations and semantic understanding given by CLIP [13]. Similarly, in [14], the authors incorporate vision-language models into end-to-end robotic control to boost generalization and enable emergent semantic reasoning (refer to such category of models as vision-language-action models); moreover, in [15] they propose SARA-RT, a Transformer adaptation procedure, where the regular attention module is replaced by a light computationally efficient attention block.

Besides, in [16], the authors present VLMaps, a spatial map representation that directly fuses pretrained visual-language features with a 3D reconstruction of the physical world, and then use large language models (LLMs) to create open-vocabulary navigation goals from a list of specific actions; likewise, in [17] the authors present the embodied language model to introduce the time parameter to visual language models and show their application for understanding of driving scenes with large spatial and temporal spans.

Furthermore, although these deep learning models attempt to solve a single problem, a general multitask data-driven model is DeepMind Gato [18], a single generalist agent where "the same network with the same weights can play Atari, caption images, chat, stack blocks with a real robot arm and much more, deciding based on its context whether to output text, joint torques, button presses, or other tokens."

However, based on the works of [19, 20] on scaling law trends, in [13] the authors raise several concerns, not only considering the size of the dataset and the amount of computing used for training but also in the bias that deep learning models might introduce through generalization in large datasets; regarding domestic robots, this bias might result in a single way to perform a task without considering users' will—task performance generalization in personal robots might blur individual intentions and expectations and cause not only annoyance and frustration but also accidents and serious injuries.

General-purpose command understanding for service robots using LLMs has been proposed in several works recently. With TidyBot [21], the authors show that the summarization capabilities of LLMs can be used to generalize user preferences for domestic robots performing cleaning up tasks (e.g., from "Put the banana in the red container" it can be generalized to deposit all fruits in that container) while in [22] the authors present OK-Robot, an open knowledge-based robotics framework offering an integrated solution for pick-and-drop operations by combining pretrained vision, manipulation, and navigation models. Also, in [23], the authors present CodeBotler, where they use LLMs to convert natural language task descrip-

tions into programs in an embedded domain-specific language (eDSL) in Python to abstract key robot skills.

To assess such systems, it has been necessary to create new benchmarks, such as the aforementioned robot competitions for domestic service robots; likewise, in [23], the authors propose RoboEval, a benchmark for evaluating LLMs' capabilities of generating programs to complete service robot tasks from multiple natural language prompts per task by checking execution traces starting with multiple initial states, and checking whether the traces satisfy temporal logic properties that encode correctness for each task. Correspondingly, in [24] the authors propose PlanBench, an extensible benchmark suite based on two different international planning competition domains (Blocksworld and Logistics) to test the capabilities of LLMs in planning or reasoning about actions and change, with sufficient diversity in both the task domains and the specific planning capabilities through around 26,250 prompts.

However, most of these systems highly rely on LLMs to create the plans (or even the code to perform such plans) but it has been demonstrated the limitations of such approaches in task-specific problems [25] and we have found it also inaccurate in natural speech commands; also in [24] the authors highlight that it becomes hard to tell whether LLMs are planning or merely retrieving from their vast world knowledge and their studies show that LLM performance falls short on many critical capabilities, including plan generation, even with the SOTA models. Similarly, in [23], the authors express that their system CodeBotler does not support low-level behaviors and generates robot plans in an open-loop fashion, rendering it incapable of reacting to unexpected changes in the environment and the users still need to manually specify constraints for each task.

It is important to highlight that current LLM-based systems really depend on the request input to such models. For example, VLMaps provides a list of example command sentences and how the model should interpret them:

```
"""
# move back and forth between the chair and the
    table 3 times
pos1 = robot.get_pos('chair')
pos2 = robot.get_pos('table')
for i in range(3):
    robot.move_to(pos1)
    robot.move_to(pos2)
"""
```

Similarly, CodeBotler provides the LLM with a task description and what should be the output instruction to it:

```
"""
# Pick up an object if you are not already holding
    one. You can only hold one object at a time.
def pick(obj: str) -> None
"""
```

In PlanBench, the authors provide a complex description of the robot's skills and the restrictions and limitations; then, they provide a clear description of the current status of the environment and finally request a goal.

```
"""
I am playing with a set of blocks where I need to
    arrange the blocks into stacks.
Here are the actions I can do

    Pick up a block
    Unstack a block from on top of another block
    ...

I have the following restrictions on my actions:
    I can only pick up or unstack one block at a
    time.
    I can only pick up or unstack a block if my
    hand is empty.
    ...

[STATEMENT]
As initial conditions I have that, the red block
    is clear, the blue block is clear...

My goal is to have the orange block on top of the
    blue block.
"""
```

LLMs are isolated language models that lack sensors and actuators, and therefore deeply depend on the input sentences to know the current internal state of the robot and the surrounding environment and the actions provided are open-loop responses. This high specificity in the input requests can be seen as rules in expert system architectures; in this sense, [26] assesses visual language models' ability to reason based on visual clues where their analysis reveals that perceptual understanding may be the main bottleneck, as the models perform better when provided with appropriate textual descriptions; in the same way, [27] studies the performance of multiple algorithms that learn from feedback on improving LLM reasoning capabilities with the Expert Iteration algorithm performing best in most cases.

Accordingly, instead of letting LLMs generate a plan through a very long and complex input, we have proposed a framework to parse GPSR commands using a syntactic parser to create a conceptual dependency (CD) structure that allows a rule-based expert system to generate a global plan that the robot can execute. CD theory [28] asserts that an action is the base of any sentence, and from the verb, we can construct a structure that describes its meaning; derived from this, a finite number of primitives compose the key units to construct and understand more complex sentences where any CD primitive contains some of the following elements: ACTOR, ACTION, OBJECT, LOCATION, STATE, and TIME.

In this context, we extend these approaches by introducing a LLM to simplify the input command by removing ambiguities and unnecessary information and rephrasing the expression to improve the semantic understanding; by doing so, we add structure to a general command that can be used later to generate plans by using complex prompts with fixed layouts as input. Specifically, to address this problem, in [29], we classify the [EE]GPSR commands from the RoboCup competition into several categories:

- Follow person commands (FP)
- Guide person commands (GP)
- Gender person commands (GeP)

Table 1. Robot's skills for type of command (from [29]).

	SR	NV	PR	OR	OM
FP	○	○	○	···	···
GP	○	○	○	···	···
GeP	○	○	○	···	···
PP	○	○	○	···	···
NP	○	○	○	···	···
HMP	○	○	○	···	···
PI	○	○	○	···	···
HMO	○	○	···	○	···
FO	○	○	···	○	···
BMO	○	○	···	○	○
PO	○	○	···	○	○
HO	○	○	○	○	○

- Person pose commands (PP)
- Name person commands (NP)
- How many people commands (HMP)
- Person instructions commands (PI)
- How many objects commands (HMO)
- Feature object commands (FO)
- Bring me object commands (BMO)
- Place object commands (PO)
- Handover object commands (HO)

where each command category is executed as a combination of several skills, namely, speech recognition (SR), navigation (NV), person recognition (PR), object recognition (OR), and object manipulation (OM), as shown in **Table 1**.

Then, we solve complex action planning by combining general-purpose service robot models based on expert systems (as in [29]) with standard and data-driven skill models (e.g., CLIPort) and then, evaluate them in real scenarios explicitly designed to test such systems, like robot competitions. Furthermore, although deep learning systems generalize well, they should serve to create local plans and their execution should still be performed considering human intentions and using active reasoning as in [30–32] to encompass the specificities of the current scenario and to handle unexpected behaviours.

In a nutshell, a syntactic parser receives the command and classifies each word according to a class (e.g., noun, verb, article, direct/indirect object); then, a semantic module compares the command to several conceptual dependency primitives. For example, a motion primitive can be represented as:

(PTRANS (ACTOR NIL) (OBJECT NIL) (FROM NIL) (TO NIL))



Fig. 3. We use Toyota's HSR [36] to perform the commands and the WRS arena setup.

The motion command "Take this apple to the able," where *able* was misunderstood and therefore discarded in the syntactic process, the semantic module would give:

(PTRANS (ACTOR *robot*) (OBJECT *apple*) (FROM *current-robot-position*) (TO *NIL*))

and act in consequence (for example, requesting the missing information).

Given a complex command and corresponding conceptual dependencies, a global plan can be generated as a sequence of steps, each consisting of one or several skills, using a rule-based expert system (like CLIPS, "a data-driven program where the facts, and objects if desired, are the data that stimulate execution via the inference engine" [33,34]). Each step is observed by an active reasoning system to perform the subtasks and solve the problems that might arise without replanning all the tasks (e.g., dynamic obstacle avoidance or retaking an object after dropping it).

5. Results and Discussion

We evaluated the natural language understanding in [35], where we randomly generated 50 commands using the RoboCup's GPSR command generator as in https://github.com/kyordhel/GPSRCmdGen (**Fig. 3**); the system achieved 95% for the easiest level, 54% for the medium level, and 72% for the hardest level leading to a global accuracy of 82% in tagging the elements and making the CD structures.

Taking into account these results, it is evident that with more sophisticated nouns and commands we need an extra step to solve these kinds of complications or ambiguities. Therefore, we propose using LLM to rephrase the input command to a more structured sentence useful for conceptual dependencies' construction. In our experiments, we use ChatGPT [h], a natural language processing (NLP) system developed by OpenAI, because it is a powerful NLP system able to understand the context of human conversation and generate appropriate responses where the system achieved 100% for the easiest level, 100% for the medium level, and 88% for the hardest level, leading to a global ac-

curacy of 96%. Some commands that could not be understood are the following:

- Take out the garbage. → Guide the garbage out.
- Describe the objects on the table. → Find the objects on the table and describe them.

Then, we describe the planning process as in [37] to shed some light on the needs that an autonomous system might require. Here, we are using CLIPS for the inference process and plan generation; this system relies on three key components: 1) templates, 2) rules, and 3) facts. While a template follows a standard definition as a structure that encompasses the definition of a class and facts are instances of those templates, in an inference system such as CLIPS, the idea of a rule goes beyond a simple state machine because they consider past, present, and future interactions where a rule may activate or cancel other rules, therefore, resulting in complex planning; in a recent book by CLIPS author, he shows how to create a fully functional text adventure game and, in the end, we can consider a general-purpose service robot as a player in a role-playing game. Next, we will describe our implementation and main findings.

We will show some simple definitions to answer the question "Where is <NAME>?" and then we will generalize them to perform complex commands. First, a fact template might look like:

```
(deftemplate item
  ( field type (type SYMBOL)(default nil) )
  ( field name (type SYMBOL)(default nil) )
  ( field num (type NUMBER)(default 1) )
  ( field room (type SYMBOL)(default nil) )
  ( field zone (type SYMBOL)(default nil) )
  ( multifield objs (type SYMBOL)(default nil) )
  ( multifield pose (type NUMBER)(default 0 0 0) )
)
```

Next, we need to declare a conceptual dependency template to be filled in the NLP module:

```
(deftemplate qtrans
  ( field actor (type SYMBOL)(default nil) )
  ( field question (type SYMBOL)(default nil) )
  ( field name (type SYMBOL)(default nil) )
)
```

and a rule that activates if the person's fact is in the memory and the qtrans is entered from the speech module:

```
(defrule qtrans-human
  ?f <- (qtrans (name ?human)(question where))
   (item (type Human)(name ?human)(room ?room-human)(zone ?zone-human))
  =>
  (retract ?f)
   (printout t "(plan (name state)(actions answer where human " ?human " " ?room-human " " ?zone-human "))"
crlf)
)
```

For example, considering a fact declaration:

(item (type Human)(name Mark)(num 1)(room living room)(zone desk)(pose 0.0 0.0 0.0))

to attend to the request "*Where is Mark?*" from the speech recognition module, it should generate the following conceptual dependency:

(qtrans (obj Mark)(question where))

and, after the corresponding rule has been activated, CLIPS should return:

(plan (name state)(actions answer where human Mark studio desk))

Following this simple yet powerful design strategy, we have implemented several fact templates and rules that activate when a conceptual dependency appears. We can perform basic commands such as "*Go to the living room*," "*Find Mark*," or "*Give some chips to Mark*"; however, a strength of this approach relies on the experts' experience to perform a task optimally. For example, a more complex command formed by a combination of simple commands

"*Go to the living room, find Mark, and give him some chips*"

would generate three conceptual dependencies at once:

(ptrans((actor robot)(obj robot)(from robot place)(to living room)))
(attend((actor robot)(obj Mark)))
(atrans((actor robot)(obj chips)(from chips place)(to Mark)))

A standard planning system would perform each of these rules separately, in sequence, which might not result in an optimal plan; for example, in the case that the object (the chips) is in a different room than in the target location (the living room), performing these task individually would result in the robot going to the target location, finding the person, going to a different room to take the object, going back to the living room, finding the person again, and finally, delivering the object. An optimal rule-based plan will take the object first, then go to the target location, find the person, and deliver the object.

Another strength of these rule-based systems is dealing with incomplete information. A conceptual dependency has a piece of minimum information to be performed and, therefore, when incomplete commands are detected, the robot might ask for the missing data. The command

"*Give an apple*"

would generate the conceptual dependency "(atrans (actor robot)(obj apple)(to nil))," and activate a rule to complete the command, like the following:

```
(defrule exec-atrans-no-recipient
  (atrans (actor ?actor)(to nil))
  =>
   (printout t "(plan (name state)(actions ask human recipient))" crlf)
)
```

It is important to consider that, while a conceptual dependency has not been attended to, it remains in memory and therefore, after the missing information has been provided, the desired plan is generated. Finally, in our experiments, we have provided several instances of similar instruction sets at the same time, resulting in very long commands where the inference system has been able to process them, e.g.,

First, go to the office, find Paola, and give her an apple; then, go to the living room, find Luis, and give him an orange. Finally, go to the kitchen.

Our system first divides it into several conceptual dependencies and then tests several groups and sub-groups of consecutive conceptual dependencies to make a plan. In this case, three sub-groups would be generated: two go–find–give and one go command.

Moreover, it should be noted that, whilst the number of skills increases (conveyed in the form of conceptual dependencies), the complexity of a requested command will increase. In our approach, a single command can be any combination with repetition of any conceptual dependency, and, therefore, expert experience is needed to help identify those combinations that require special attention and leave the remaining to be performed sequentially. In the following link https://youtu.be/EXz1g6tuw78 our system implementation in a domestic service robot for general-purpose tasks can be seen.

6. Conclusion

The different nature of these competitions is a consequence of the vast variety of services a service robot can provide, and therefore getting a consensus on the evaluation of a service robot's performance becomes a great challenge. In this work, through the latest works on service robots, we have described a general-purpose task for a personal domestic robot. The performance of a single command should comprise several skills and the robot should deal with incomplete information, environmental reasoning, feature detection, dynamic physical states, and natural human-robot interaction in both directions. Several systems have been developed towards this goal; on the one hand, a single system can solve similar tasks by semantic context understanding (such as moving objects from one place to another), and, on the other hand, domestic environments that simulate physical states, allow virtual reality interfaces and support object's semantic information has been developed. However, a planning module is necessary to concatenate a series of actions to attend to complex commands. Here we propose to look back to expert systems to close the gap toward general-purpose service robots and to explore new ways to apply them to deep-learning-based models.

References:

[1] L. Contreras and H. Okada, "Personal and Domestic Robotics," Encyclopedia of Robotics, 2022.

[2] T. Inamura and Y. Mizuchi, "SIGVerse: A Cloud-Based VR Platform for Research on Multimodal Human-Robot Interaction," Front. Robot. AI, Vol.8, Article No.549360, 2021. https://doi.org/10.3389/frobt.2021.549360

[3] L. Contreras, T. Yamamoto, Y. Matsusaka, and H. Okada, "Towards general purpose service robots: World Robot Summit – Partner Robot Challenge," Advanced Robotics, Vol.36, Nos.17-18, pp. 812-824, 2022. https://doi.org/10.1080/01691864.2022.2109428

[4] S. James, Z. Ma, D. R. Arrojo, and A. J. Davison, "RLBench: The Robot Learning Benchmark and Learning Environment," IEEE Robotics and Automation Letters, Vol.5, Issue 2, pp. 3019-3026, 2020. https://doi.org/10.1109/LRA.2020.2974707

[5] A. Zeng, P. Florence, J. Tompson, S. Welker, J. Chien, M. Attarian, T. Armstrong, I. Krasin, D. Duong, V. Sindhwani, and J. Lee, "Transporter Networks: Rearranging the Visual World for Robotic Manipulation," Conf. on Robot Learning (CoRL), 2020.

[6] V. Kumar, R. Shah, G. Zhou, V. Moens, V. Caggiano, J. Vakil, A. Gupta, and A. Rajeswaran, "RoboHive: A Unified Framework for Robot Learning," NeurIPS: Conf. on Neural Information Processing Systems, 2023.

[7] A. Szot, A. Clegg, E. Undersander, E. Wijmans, Y. Zhao, J. Turner, N. Maestre, M. Mukadam, D. S. Chaplot, O. Maksymets, A. Gokaslan, V. Vondrus, S. Dharur, F. Meier, W. Galuba, A. Chang, Z. Kira, V. Koltun, J. Malik, M. Savva, and D. Batra, "Habitat 2.0: Training Home Assistants to Rearrange their Habitat," Proc. of the 35th Int. Conf. on Neural Information Processing Systems (NIPS'21), Article No.20, pp. 251-266, 2021.

[8] X. Puig, E. Undersander, A. Szot, M. D. Cote, T.-Y. Yang, R. Partsey, R. Desai, A. W. Clegg, M. Hlavac, S. Y. Min, V. Vondrus, T. Gervet, V.-P. Berges, J. M. Turner, O. Maksymets, Z. Kira, M. Kalakrishnan, J. Malik, D. S. Chaplot, U. Jain, D. Batra, A. Rai, and R. Mottaghi, "Habitat 3.0: A Co-Habitat for Humans, Avatars and Robots," arXiv preprint, arXiv:2310.13724, 2023. https://doi.org/10.48550/arXiv.2310.13724

[9] L. Contreras, Y. Matsusaka, T. Yamamoto, and H. Okada, "sD-SPL – Towards a benchmark for general-purpose task evaluation in domestic service robots," The 39th Annual Conf. of the Robotics Society of Japan, 2021.

[10] S. Srivastava, C. Li, M. Lingelbach, R. Martín-Martín, F. Xia, K. Vainio, Z. Lian, C. Gokmen, S. Buch, C. K. Liu, S. Savarese, H. Gweon, J. Wu, and L. Fei-Fei, "BEHAVIOR: Benchmark for Everyday Household Activities in Virtual, Interactive, and Ecological Environments," Conf. on Robot Learning (CoRL), 2021.

[11] C. Li, F. Xia, R. Martín-Martín, M. Lingelbach, S. Srivastava, B. Shen, K. Vainio, C. Gokmen, G. Dharan, T. Jain, A. Kurenkov, C. K. Liu, H. Gweon, J. Wu, L. Fei-Fei, and S. Savarese, "iGibson 2.0: Object-Centric Simulation for Robot Learning of Everyday Household Tasks," Conf. on Robot Learning (CoRL), 2021.

[12] M. Shridhar, L. Manuelli, and D. Fox, "CLIPort: What and Where Pathways for Robotic Manipulation," Proc. of the 5th Conf. on Robot Learning (CoRL), 2021.

[13] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning Transferable Visual Models From Natural Language Supervision," Proc. of the 38th Int. Conf. on Machine Learning, 2021.

[14] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, P. Florence, C. Fu, M. G. Arenas, K. Gopalakrishnan, K. Han, K. Hausman, A. Herzog, J. Hsu, B. Ichter, A. Irpan, N. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, L. Lee, T.-W. E. Lee, S. Levine, Y. Lu, H. Michalewski, I. Mordatch, K. Pertsch, K. Rao, K. Reymann, M. Ryoo, G. Salazar, P. Sanketi, P. Sermanet, J. Singh, A. Singh, R. Soricut, H. Tran, V. Vanhoucke, Q. Vuong, A. Wahid, S. Welker, P. Wohlhart, J. Wu, F. Xia, T.

Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich, "RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control," arXiv preprint, arXiv:2307.15818, 2023. https://doi.org/10.48550/arXiv.2307.15818

[15] I. Leal, K. Choromanski, D. Jain, A. Dubey, J. Varley, M. Ryoo, Y. Lu, F. Liu, V. Sindhwani, Q. Vuong, T. Sarlos, K. Oslund, K. Hausman, and K. Rao, "SARA-RT: Scaling up Robotics Transformers with Self-Adaptive Robust Attention," 2024 IEEE Int. Conf. on Robotics and Automation (ICRA), 2024. https://doi.org/10.1109/ICRA57147.2024.10611597

[16] C. Huang, O. Mees, A. Zeng, and W. Burgard, "Visual Language Maps for Robot Navigation," 2023 IEEE Int. Conf. on Robotics and Automation (ICRA), 2023. https://doi.org/10.1109/ICRA48891.2023.10160969

[17] Y. Zhou, L. Huang, Q. Bu, J. Zeng, T. Li, H. Qiu, H. Zhu, M. Guo, Y. Qiao, and H. Li, "Embodied Understanding of Driving Scenarios," arXiv preprint, arXiv:2403.04593, 2024. https://doi.org/10.48550/arXiv.2403.04593

[18] S. Reed, K. Zolna, E. Parisotto, S. G. Colmenarejo, A. Novikov, G. Barth-Maron, M. Gimenez, Y. Sulsky, J. Kay, J. T. Springenberg, T. Eccles, J. Bruce, A. Razavi, A. Edwards, N. Heess, Y. Chen, R. Hadsell, O. Vinyals, M. Bordbar, and N. de Freitas, "A Generalist Agent," arXiv preprint, arXiv:2205.06175, 2022. https://doi.org/10.48550/arXiv.2205.06175

[19] J. Hestness, S. Narang, N. Ardalani, G. Diamos, H. Jun, H. Kianinejad, Md. M. A. Patwary, Y. Yang, and Y. Zhou, "Deep Learning Scaling is Predictable, Empirically," arXiv preprint, arXiv:1712.00409, 2017. https://doi.org/10.48550/arXiv.1712.00409

[20] J. Kaplan, S. McCandlish, T. Henighan, T. B. Brown, B. Chess, R. Child, S. Gray, A. Radford, J. Wu, and D. Amodei, "Scaling Laws for Neural Language Models," arXiv preprint, arXiv:2001.08361, 2020. https://doi.org/10.48550/arXiv.2001.08361

[21] J. Wu, R. Antonova, A. Kan, M. Lepert, A. Zeng, S. Song, J. Bohg, S. Rusinkiewicz, and T. Funkhouser, "TidyBot: Personalized Robot Assistance with Large Language Models," IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2023. https://doi.org/10.1109/IROS55552.2023.10341577

[22] P. Liu, Y. Orru, J. Vakil, C. Paxton, N. M. M. Shafiullah, and L. Pinto, "OK-Robot: What Really Matters in Integrating Open-Knowledge Models for Robotics," arXiv preprint, arXiv:2401.12202, 2024. https://doi.org/10.48550/arXiv.2401.12202

[23] Z. Hu, F. Lucchetti, C. Schlesinger, Y. Saxena, A. Freeman, S. Modak, A. Guha, and J. Biswas, "Deploying and Evaluating LLMs to Program Service Mobile Robots," IEEE Robotics and Automation Letters, Vol.9, Issue 3, pp. 2853-2860, 2024. https://doi.org/10.1109/LRA.2024.3360020

[24] K. Valmeekam, M. Marquez, A. Olmo, S. Sreedharan, and S. Kambhampati, "PlanBench: An Extensible Benchmark for Evaluating Large Language Models on Planning and Reasoning about Change," arXiv preprint, arXiv:2206.10498, 2023. https://doi.org/10.48550/arXiv.2206.10498

[25] K. Valmeekam, A. Olmo, S. Sreedharan, and S. Kambhampati, "Large Language Models Still Can't Plan (A Benchmark for LLMs on Planning and Reasoning about Change)," 2022.

[26] Y. Zhang, H. Bai, R. Zhang, J. Gu, S. Zhai, J. Susskind, and N. Jaitly, "How Far Are We from Intelligent Visual Deductive Reasoning?," arXiv preprint, arXiv:2403.04732, 2024. https://doi.org/10.48550/arXiv.2403.04732

[27] A. Havrilla, Y. Du, S. C. Raparthy, C. Nalmpantis, J. D.-Yu, M. Zhuravinskyi, E. Hambro, S. Sukhbaatar, and R. Raileanu, "Teaching Large Language Models to Reason with Reinforcement Learning," arXiv preprint, arXiv:2403.04642, 2024. https://doi.org/10.48550/arXiv.2403.04642

[28] R. C. Schank, "Conceptual dependency: A theory of natural language understanding," Cognitive Psychology, Vol.3, Issue 4, pp. 552-631, 1972. https://doi.org/10.1016/0010-0285(72)90022-9

[29] J. Savage, D. Rosenblueth, M. Matamoros, M. Negrete, L. Contreras, J. Cruz, R. Martell, H. Estrada, and H. Okada, "Semantic reasoning in service robots using expert systems," Robotics and Autonomous Systems, Vol.114, pp. 77-92, 2019. https://doi.org/10.1016/j.robot.2019.01.007

[30] E. Vazquez, L. Contreras, H. Okada, and N. Iwahashi, "Multimodal human intention-driven robot motion control in collaborative tasks," The 38th Annual Conf. of the Robotics Society of Japan, 2020.

[31] L. Contreras, Y. Muto, H. Okada, and A. Sakamaki, "Robot-object interaction strategies for object recognition," The 37th Annual Conf. of the Robotics Society of Japan, 2019.

[32] L. Contreras, H. Yokoyama, and H. Okada, "Multimodal feedback for active perception in service robots," IEEE/RSJ IROS Workshops, 2018.

[33] G. Riley, "CLIPS Reference Manual Version 6.0. Technical Report Number JSC-25012," Software Technology Branch, Lyndon B. Johnson Space Center, 1994.

[34] G. Riley, "Adventures in Rule-Based Programming: A CLIPS Tutorial," Secret Society Software, LLC, 2022.

[35] S. Ortuno-Chanelo, L. Contreras, J. Savage, and H. Okada, "Keep it Simple: Understanding Natural Language Commands for General-Purpose Service Robots," IEEE/SICE Int. Symp. on System Integration, 2024. https://doi.org/10.1109/SII58957.2024.10417341

[36] T. Yamamoto, K. Terada, A. Ochiai, F. Saito, Y. Asahara, and K. Murase, "Development of human support robot as the research platform of a domestic mobile manipulator," ROBOMECH J., Vol.6, Article No.4, 2019. https://doi.org/10.1186/s40648-019-0132-3

[37] L. Contreras, J. Savage, S. Ortuno-Chanelo, M. Negrete, A. Sakamaki, and H. Okada, "Fail It till You Make It: Error Expectation in Complex-Plan Execution for Service Robots," Int. Conf. on Interactive Collaborative Robotics, pp. 36-46, 2023. https://doi.org/10.1007/978-3-031-43111-1_4

Supporting Online Materials:

[a] RoboCup@Home. https://athome.robocup.org/ [Accessed February 4, 2024]

[b] ERL. https://eu-robotics.net/european-robotics-league/ [Accessed February 4, 2024]

[c] ERL Consumer Service Robots. https://eu-robotics.net/erl-professional/ [Accessed February 4, 2024]

[d] RockIn@Home. http://rockinrobotchallenge.eu/home.php [Accessed February 4, 2024]

[e] ERL Smart Cities. https://eu-robotics.net/erl-smart-cities/ [Accessed February 4, 2024]

[f] Roboethics Competition. https://competition.raiselab.ca/ [Accessed February 4, 2024]

[g] RoboCup@Home Command Generator. https://github.com/kyordhel/GPSRCmdGen [Accessed February 4, 2024]

[h] ChatGPT OpenAI. https://openai.com/chatgpt [Accessed February 4, 2024]



Name:
Luis Contreras

ORCID:
0000-0002-7957-4941

Affiliation:
Tokyo Information Design Professional University

Address:
2-7-1 Komatsugawa, Edogawa-ku, Tokyo 132-0034, Japan

Brief Biographical History:
2017 Received Ph.D. degree in Computer Science from Visual Information Laboratory, Department of Computer Vision, University of Bristol
2017- Bio-robotics Laboratory, School of Engineering, National Autonomous University of Mexico
2018- Advanced Intelligence & Robotics Research Center, Tamagawa University
2024- Adjunct Researcher Fellow, Tokyo Information Design Professional University

Main Works:
• S. Ortuno-Chanelo, L. Contreras, J. Savage, and H. Okada, "Keep it simple: Understanding natural language commands for general-purpose service robots," IEEE/SICE Int. Symp. on System Integration (SII), 2024.
• L. Contreras, T. Yamamoto, Y. Matsusaka, and H. Okada, "Towards general purpose service robots: World Robot Summit – Partner Robot Challenge," Advanced Robotics, Vol.36, Nos.17-18, pp. 812-824, 2022.
• J. Savage, D. A. Rosenblueth, M. Matamoros, M. Negrete, L. Contreras, J. Cruz, R. Martell, H. Estrada, and H. Okada, "Semantic reasoning in service robots using expert systems," Robotics and Autonomous Systems, Vol.114, pp. 77-92, 2019.
• M. Negrete, J. Savage, and L. Contreras, "A motion planning system for a domestic service robot," Informatics and Automation (Proc. of SPIIRAS), Issue 60, pp. 5-38, 2018.
• L. Contreras and W. Mayol-Cuevas, "O-POCO: Online point cloud compression mapping for visual odometry and SLAM," IEEE Int. Conf. on Robotics and Automation (ICRA), 2017.

Membership in Academic Societies:
• Robotics Mexican Federation, Fellow
• RoboCup Federation, Executive Committee
• The Robotics Society of Japan (RSJ)
• Institute of Electrical and Electronics Engineers (IEEE) Robotics and Automation Society (RAS)



Name:
Jesús Savage

ORCID:
0000-0002-6340-8375

Affiliation:
Electrical Engineering Department, National Autonomous University of Mexico

Address:
Circuito Exterior S/N, Ciudad Universitaria, Coyoacán, Mexico City 04510, Mexico

Brief Biographical History:
1985 Received B.S. degree in Computer Engineering from National Autonomous University of Mexico
1990 Received M.S. degree in Electrical Engineering from National Autonomous University of Mexico (Part of the Master Thesis Developed at The University of California)
1995 Received Ph.D. degree in Electrical Engineering from University of Washington
1995- Professor, Electrical Engineering Department, School of Engineering, National Autonomous University of Mexico
2003- Founder, Bio-Robotics Laboratory, National Autonomous University of Mexico

Main Works:
• O. Fuentes, J. Savage, and L. Contreras, "A SLAM system based on Hidden Markov Models," Informatics and Automation, Vol.21, No.1, pp. 181-212, 2022. https://doi.org/10.15622/ia.2022.21.7
• J. Hernández, J. Savage, M. Negrete, L. Contreras, C. Sarmiento, O. Fuentes, and H. Okada, "Sparse-Map: automatic topological map creation via unsupervised learning techniques," Advanced Robotics, Vol.36, Issues 17-18, pp. 825-835, 2022. https://doi.org/10.1080/01691864.2022.2114296
• A. Nakayama, D. Ruelas, J. Savage, and E. Bribiesca, "Teleoperated Service Robot with an Immersive Mixed Reality Interface," Informatics and Automation, Vol.20, No.6, pp. 1187-1223, 2021. http://www.proceedings.spiiras.nw.ru/index.php/sp/article/view/15069
• C. Sarmiento and J. Savage, "Comparison of Two Objects Classification Techniques using Hidden Markov Model and Convolutional Neural Networks," Informatics and Automation, Vol.19, No.6, pp. 1222-1254, 2020. https://doi.org/10.15622/ia.2020.19.6.4
• J. Savage, D. Rosenblueth, M. Matamoros, M. Negrete, L. Contreras, J. Cruz, R. Martell, H. Estrada, and H. Okada, "Semantic reasoning in service robots using expert systems," Robotics and Autonomous Systems, Vol.114, pp. 77-92, 2019.

Membership in Academic Societies:
• Robotics Mexican Federation, President (2010-2012)
• RoboCup 2012 in Mexico City, Co-Chair
• RoboCup Federation, Board of Trustees
• Institute of Electrical and Electronics Engineers (IEEE) Robotics and Automation Society (RAS)



Name:
Stephany Ortuño-Chanelo

ORCID:
0000-0002-3671-0632

Affiliation:
Multimodal Applied Learning Laboratory, Polytechnic University of Turin

Address:
Torino 10129, Italy

Brief Biographical History:
2019-2021 Master Student of Electrical Engineering, Faculty of Engineering, National Autonomous University of Mexico
2023- Ph.D. Candidate, National Ph.D. in Artificial Intelligence, Politecnico di Torino
2023- Ph.D. Industrial Collaborator, COMAU

Main Works:
• S. Ortuno-Chanelo, L. Contreras, J. Savage, and H. Okada, "Keep it Simple: Understanding Natural Language Commands for General-Purpose Service Robots," IEEE/SICE Int. Symp. on System Integration, 2024.
• L. Contreras, J. Savage, S. Ortuno-Chanelo, M. Negrete, A. Sakamaki, and H. Okada, "Fail It till You Make It: Error Expectation in Complex-Plan Execution for Service Robots," Int. Conf. on Interactive Collaborative Robotics, pp. 36-46, 2023.
• A. Presacco, J. Savage, J. Cruz, and S. Ortuño, "A framework to process natural speech in service robots using a combination of a speech recognition system, universal dependencies extracted by means of the Stanford parser and an expert system," Int. Conf. on Interactive Collaborative Robotics, pp. 172-181, 2021.



Name:
Marco Negrete

ORCID:
0000-0002-5468-2807

Affiliation:
Signal Processing Department, National Autonomous University of Mexico

Address:
Circuito Exterior S/N, Ciudad Universitaria, Coyoacán, Mexico City 04510, Mexico

Brief Biographical History:
2017- Associate Professor, School of Engineering, National Autonomous University of Mexico
2019 Received Ph.D. degree in Computer Science from National Autonomous University of Mexico

Main Works:
• H. Avilés, M. Negrete, A. Reyes, R. Machucho, K. Rivera, G. de-la-Garza, and A. Petrilli, "Autonomous Behavior Selection For Self-driving Cars Using Probabilistic Logic Factored Markov Decision Processes," Applied Artificial Intelligence, Vol.38, Issue 1, Article No.2304942, 2024. https://doi.org/10.1080/08839514.2024.2304942
• H. Avilés, M. Negrete, R. Machucho, K. Rivera, D. Trejo, and H. Vargas, "Probabilistic Logic Markov Decision Processes for Modeling Driving Behaviors in Self-driving Cars," A. C. B. Garcia, M. Ferro, and J. C. R. Ribón (Eds.), "Advances in Artificial Intelligence – IBERAMIA 2022," Lecture Notes in Computer Science (LNCS), Vol.13788, pp. 366-377, 2022. https://doi.org/10.1007/978-3-031-22419-5_31
• M. Negrete, J. Savage, and L. Contreras, "A motion planning system for a domestic service robot," Informatics and Automation (Proc. of SPIIRAS), Issue 60, pp. 5-38, 2018.
• Automatic control, self-driving cars and behavioral sciences.

Membership in Academic Societies:
• Mexican Robotics Federation, Board of Directors
• National System of Researchers



Name:
Hiroyuki Okada

ORCID:
0000-0001-8422-0542

Affiliation:
Professor, Tokyo Information Design Professional University

Address:
2-7-1 Komatsugawa, Edogawa-ku, Tokyo 132-0034, Japan

Brief Biographical History:
1986-2002 Fujitsu Laboratories Ltd.
2002-2006 Tokai University
2006-2024 Tamagawa University
2024- Tokyo Information Design Professional University

Main Works:
• H. Okada, T. Inamura, and K. Wada, "What competitions were conducted in the service categories of the World Robot Summit?," Advanced Robotics, Vol.33, Issue 17, pp. 900-910, 2019. https://doi.org/10.1080/01691864.2019.1663608

Membership in Academic Societies:
• Japanese Cognitive Science Society (JCSS)
• The Robotics Society of Japan (RSJ)
• The Japanese Society for Artificial Intelligence (JSAI)
• Institute of Electrical and Electronics Engineers (IEEE) Robotics and Automation Society (RAS)
• RoboCup Federation, Board of Trustees