

# Steps Towards Generalized Manipulation Action Plans - Tackling Mixing Task

Vanessa Hassouna<sup>1,\*</sup>, Alina Hawkin<sup>1,\*</sup> and Michael Beetz<sup>1</sup>

<sup>1</sup>*Institute for Artificial Intelligence, University of Bremen, Bremen, Germany*

## Abstract

In the rapidly evolving field of household robotics, the ability to autonomously execute complex tasks like "Serve Me Breakfast" introduces considerable technical challenges, particularly in the precise and adaptive handling of mixing tasks that involve substances with diverse densities and viscosities. This study advances the field of cognitive robotics by delving into the intricacies of everyday, complex, and highly variable activities within household settings. These scenarios demand that robots manage multiple, interconnected actions—a concept known as schemas. We present a holistic strategy to enhance household robots with sophisticated task execution capabilities, utilizing the Cognitive Robotic Abstract Machine (CRAM) framework for strategic planning and execution. Our method begins with an in-depth analysis of human behaviour to develop a theoretical model that guides the creation of adaptable and comprehensible action plans for robots. A crucial element of our approach involves using Narrative Enabled Episodic Memories (NEEMs), which capture detailed records of task executions to aid performance analysis and experiential learning. We propose incorporating additional criteria based on events recorded in the NEEMs to assess task success. For instance, a robot maintaining a steady grip on the whisk while interacting with a fluid suggests correctly executing mixing tasks. These criteria enable further evaluation of performance through simulations, despite potential limitations in simulation fidelity. This article explores the transition from human expertise to robotic execution of mixing tasks in household environments, the methodology for gathering and analysing NEEMs, and their prospective future applications.

## Keywords

Domestic robotics, Mixing tasks, Cognitive Architecture, Undetermined Action Description, CRAM framework, Narrative Enabled Episodic Memories (NEEMs), Task parameters, Reasoning, Simulation, KnowRob

## 1. Introduction

Enabling robots to perform complex, everyday tasks, such as preparing breakfast, with an understanding and adaptability akin to human behaviour is particularly challenging. This challenge is exciting because it transcends mere mechanical execution, advancing into the realm of cognitive robotics, where robots must not only perform actions but also comprehend their purposes and contextual implications.

The challenge is significant because everyday activities in human environments are inherently complex and highly variable, requiring careful deliberation to ensure functional and adaptable robotic actions. These tasks often involve actions within actions—schemas [1] that robots must dynamically interpret and execute. Our research demonstrates that this layered understanding is possible, as shown in the context of object transportation tasks [2], where robotic actions are nested within broader goal-directed behaviours.

Less explored, however, is whether similar schema-based approaches can be applied to manipulation functions that demand a deeper understanding of the world and its dynamic properties—tasks that fundamentally alter the state of their environment. This paper focuses on one such category of actions: mixing tasks. Mixing is not only about the physical stirring or blending of substances but also involves understanding the physical and chemical properties that change due to these actions, which is a very challenging problem [3]. Our method offers a perspective by conceptualizing the mixing process as a high-level symbolic description. This approach allows for a more abstract mixing task execution without

---

*Workshop on Actionable Knowledge Representation and Reasoning for Robots (AKR<sup>3</sup>) at Extended Semantic Web Conference (ESWC), May 27, 2024, Heraklion, Crete, Greece*

\*Corresponding author.

✉ hassouna@uni-bremen.de (V. Hassouna); hawkin@uni-bremen.de (A. Hawkin); beetz@cs.uni-bremen.de (M. Beetz)

ORCID 0000-0003-1335-5698 (V. Hassouna); 0000-0003-1826-9983 (A. Hawkin); 0000-0002-7888-7444 (M. Beetz)

© 2024 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

necessitating an exact physical representation. Past research has demonstrated that such high-level plans are sufficiently detailed to instruct robots on the required actions effectively [4] [5].

This article offers insights into leveraging the Cognitive Robotic Abstract Machine (CRAM) [6] framework, which supports high-level reasoning and decision-making for robotic agents. This involves analyzing human behaviour to develop theoretical models that guide the creation of understandable and executable action plans for robots. This approach allows robots to interpret the essence of their tasks and make informed decisions during task execution. Furthermore, we propose enhancing the robot's ability to learn and adapt using Narrative-Enabled Episodic Memories (NEEMs) [7] [8] within the CRAM framework. These memories are recordings, detailed logs enriched with symbolic, environmental, and ontological data, providing a full perspective on each task's execution. These memories help identify the conditions necessary for successful performance, allowing for continuous improvement of robotic capabilities.

The paper's contributions include a detailed methodology for creating theoretical models essential for programming cognitive tasks in robots, introducing an undetermined action designator for executing mixing tasks, and using NEEMs to record and evaluate robot actions, enabling their skill to reason retro perspective. The paper also highlights the potential of NEEMs to advance robotic capabilities, suggesting a future with more autonomous and adaptive robotic systems across various settings.

## 2. State of the Art

Research has explored anthropomorphic-inspired approaches to enhance robotic capabilities in tasks like mixing [9]. Robots can replicate human-like movements by studying human motions and implementing inverse kinematics, fostering collaboration between humans and robots across various fields. This approach improves adaptability and opens up new possibilities in areas ranging from manufacturing to healthcare. Building upon this foundation, we introduce a high-level theoretical model for robots to simulate. Informed by human movement patterns, this model does not necessitate the robot to replicate these movements exactly. Instead, robots apply the underlying principles of these actions, optimizing performance to suit robotic capabilities and environmental constraints. Furthermore, advancements such as "BakeBot" [10] have tackled challenges in item localization and optimal mixing techniques. Robots can adapt to environmental changes while baking cookies by employing high-level motion plans and dynamic mixing trajectories. These strategies, which include compliance and force control, have proven effective in achieving desired outcomes despite potential failure scenarios.

Additional studies on human motions, such as whipping cream and tea whisking [11], provide insights into effective agitating motions and skill progression. Observations of expert confectionery hygiene experts have revealed efficient agitating motions, highlighting the importance of specific patterns like elliptical and reciprocating motions. Similarly, analysis of motions in the traditional Japanese tea ceremony has helped highlight differences in skill levels and distinct motion processes, which aid in designing theoretical models for robotic tasks like mixing. Our methodology leverages these insights to develop a nuanced motion model that can dynamically adapt to the robot's tasks.

The "Robot Household Marathon Experiment" [2] explores the complexities of underdetermined actions and dynamic environments in real-world household settings. We utilize insights from this study, applying the same framework to enhance our understanding of action execution in real-world scenarios. Although the experiment primarily demonstrated basic household tasks like setting and cleaning up a breakfast table, more than these predominantly pick-and-place actions are needed for more complex tasks such as mixing, which require learning from past experiences and adapting strategies over time. Our approach integrates advanced memory and learning capabilities, enabling robots to perform complex mixing tasks with increased autonomy and adaptability.

Regarding learning, significant research has been conducted on acquiring knowledge from human interactions within VR through NEEMs [12]. Recording VR NEEMs involves observing human actions in VR environments to generate NEEMs that can be generalized for robot actions. By extrapolating high-level actions from specific events, such as contact events and object movements, conditions for

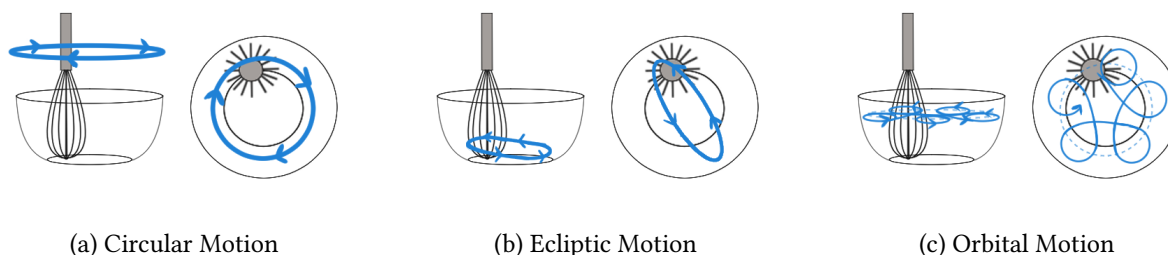
successful task completion can be inferred. This provides valuable insights into the usefulness of NEEMs for learning actions or reasoning about them. A key aspect of our approach is enabling the robot to retrospectively analyze its actions and ensure that newly introduced actions are executed successfully in different environments.

### 3. The Theoretical Planning Methodology

We begin by formulating a theoretical action model, utilizing various knowledge sources to integrate new actions into our robotic system. This model includes an Action Designator, a symbolic underdetermined descriptor that is contextualized during runtime, encompassing preconditions, task-specific requirements resolved at runtime, and postconditions that outline the intended goals [13].

#### 3.1. Observation

For the specific task of developing an action model for mixing, we used the "Max Planck Institute for Informatics Cooking Activities dataset 2.0" [14], hereafter referred to as the **dataset**. This dataset provides a collection of human-executed cooking tasks, including mixing and utilizing various tools, ingredients, and containers. We manually filtered this dataset to select videos relevant to our research based on predefined criteria. After annotating these videos, we analyzed them to identify different types of mixing actions. These actions were broken down into basic elements, such as gripping a tool or holding a bowl, which were then organized into typical phases to formulate a generalized action model. Our analysis was validated against established mixing techniques from WikiHow to ensure the accuracy of our observations [15]. WikiHow has been utilized in our other research to enhance our knowledge base and inform the design process of our model process [16].



**Figure 1:** Three Types of Mixing Motion

The evaluation process concluded with selecting 54 videos deemed relevant to our study. Initially, 273 videos were considered. This extensive review revealed that all mixing actions typically begin with the utensil in one hand and the container in the other, both ready for use. Every container is open-topped, and the utensil, usually gripped by its handle, is aimed towards the centre of the container before being inserted. Following this setup, the mixing actions vary and can be categorized into three distinct types of mixing motions:

1. **Circle motion**, rotation- drawing a circle on two axes as seen in Figure 1.a
2. **Ellipse motion**, a very simplified version would be a line - drawing on one axis back and forth as seen in Figure 1.b
3. **Orbit motion**, along an invisible big circle in even segments, small circle motions are drawn with the segments being the centre of the smaller circle as seen in Figure 1.c

We categorize the mixing motions observed in our dataset as **circular**, **ellipse**, and **orbital** mix types. The term "orbital" was inspired by the visual similarity to the moon's orbit around the sun, not implying any literal astrodynamics of orbital motion [17]. Analysis of the dataset reveals that these motion types can be repeated and linked together in sequences. However, it is important to note that more videos featuring consistent mix ingredients, tools, and containers are needed to establish any

correlation with mix duration. After the various mixing motions, a consistent final step occurs in all observed actions: the tool is withdrawn from the container. This action-to-motion strategy aligns with the Flanagan Model [18].

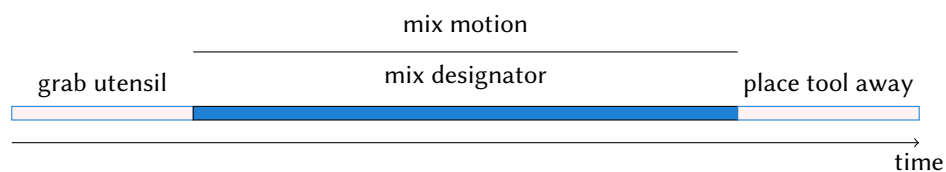
Different mixing motions are preferred by subjects in the dataset depending on the context, such as the specific ingredient being mixed and the utensil used. One limitation we encountered in the dataset was the lack of tasks involving the whipping of heavy cream or the goal of incorporating air into a mixture, which typically involves a whisk. To address this, we draw upon insights from the paper "CFD Analysis of Effective Human Motion for Whipping Heavy Cream by Hand" [11]. For incorporating air, a motion with large amplitude is necessary [11, p. 121]. The elliptical and reciprocating motions described in the paper are deemed effective. When comparing these motions with those in our dataset, we observe similarities with our defined circular and ellipse mix types. A notable difference is that the subjects in the paper also tilt the container to increase amplitude during their whipping motion without altering the whisking pattern. This adaptation suggests that our defined mix types are relevant and applicable to whipping, a specialized form of mixing.

### 3.2. The Resulting Model

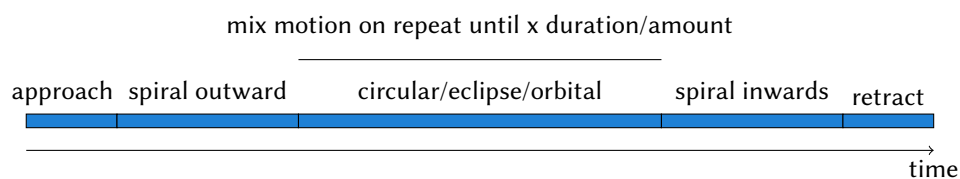
Based on our previous observations, a structured timeline for the mixing action has been developed, as shown in Figure 2. This timeline delineates the key stages involved in the mixing process:

**Pre-condition:** The initial step involves grasping the tool and the container, which is denoted as *grab utensil* in Figure 2 (left).

**Postcondition :** After completing the mixing action, the tool should be set aside, represented as *place tool away* in Figure 2a (right).



**Figure 2:** Timeline of Pre-condition and Postcondition of Mixing.



**Figure 3:** Timeline of a Mixing Designer

The mixing Designer is depicted in Figure 3. Initially, the arm equipped with the tool is guided to the container and inserted, a phase referred to as **approach**. A start position must be established to transition into repeatable circular, elliptical, orbital motions. To seamlessly connect the 'approach' to the repeatable motion, a **spiral outwards** motion is employed. Selecting the appropriate container involves careful management of the contact points with the tool. An intermediate step is implemented to prevent any unexpected collisions during the tool's withdrawal from the container. Since the 'spiral outwards' motion is confirmed to be safe, it is retraced back to its starting point before initiating a withdrawal, termed as **spiral inwards**. A **retract** phase safely concludes the mixing process.

It should be noted that the duration of the repeated mixing motions might vary based on factors such as the container's size and the mixed substance's viscosity.

## 4. Integrating New Actions into CRAM

We are developing an undetermined action designator [13] for mixing tasks within the CRAM framework. An action designator translates symbolic action descriptions into concrete ROS action goals or similar data constructs, typically through an inference engine. The effective resolution of an action designator requires defining various predicates; our Designator includes predicates such as type, context, resolution, round arms object, and source. Importantly, not all parameters need to be explicitly specified by the user; for example, if an arm is not selected, the system will autonomously attempt to resolve the Designator, possibly by determining which arm is currently available.

For the Designator to function effectively, it is essential that the tool object and container object are precisely identified. These elements are critical as they inform the system of the specific objects to interact with during the mixing process. Listing 1 presents the undetermined action designator for mixing.

Listing 1: Undetermined Mixing Action Designator

---

```
(designator action
  (type :mixing)
  (context :mix-circle)
  (resolution 12)
  (rounds 1)
  (arm (:right))
  (object ?object-container)
  (source ?object-design-source))
```

---

The next step is to resolve this high-level and abstract designator. Once resolved, it transforms into a detailed action plan for the robot, incorporating atomic actions such as approaching, each corresponding to an individual motion. To derive these atomic actions, we calculate the entire trajectory of the robot for the task in segmented increments. The functionality and effectiveness of this action designator have been extensively tested in previous research [2], demonstrating its capability to accurately and efficiently guide robotic actions in complex tasks like mixing.

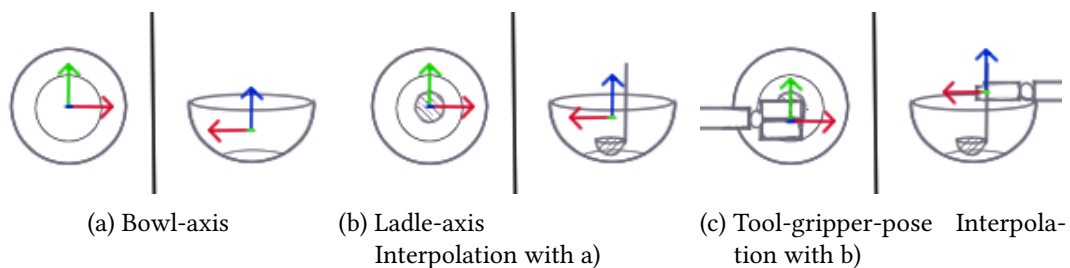


Figure 4: End Pose of Approach Pose Calculation with Container Vector Result

Once compatibility between the tool and container is established, we calculate the end pose. Consider a bowl as the container and a ladle as the tool for illustrative purposes, though this methodology applies to any tool-container combination. The desired end pose is determined by interpolating the container bottom value with the tool height. This interpolation process is depicted in Figure 4. The next step involves determining the gripper's pose on the tool and integrating this information into the previously interpolated result. Once the pose information is obtained in the object transform frame, it is converted to the base transform frame for further processing. This leads to the definition of the **approach-pose**, where the gripper positions just above the container rim.

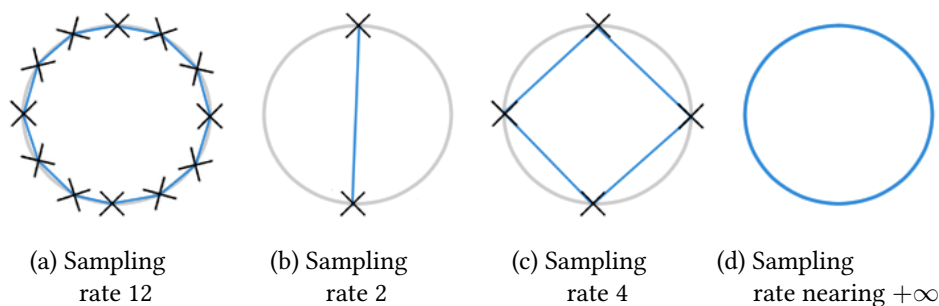
Following the approach-pose, we transition to the **start-mix-poses**, facilitating a smooth transition to the actual mixing action through a logarithmic spiral motion. The logarithmic spiral was chosen because it provides a natural motion trajectory. As the gripper remains static on the height axis during this phase, the spiral calculation is performed in **two-dimensional space**. The height dimension will

be integrated later to complete the pose configuration within the container space. The mathematical representation of a logarithmic spiral in **parametric equations** is as follows:

$$x(t) = a * e^{k*t} \cos(t) \quad (1)$$

$$y(t) = a * e^{k*t} \sin(t) \quad (2)$$

The two combined equations (1) and (2) provides the xy-coordination as a function of time  $t$ , with  $a$  representing the **section size** of the spiral, determined by the number of segments into which the spiral is divided,  $k$  is an **arbitrary constant** and  $t$  denotestime. The value  $a$  is critical, as it determines when the spiral will terminate at the maximum radius of the container. This maximum value for  $a$  is calculated based on the tool and container types, adjusting for the tool width and the distance from the tool to the inner wall of the container, as informed by our object knowledge module. This ensures that the tool remains within the container's bounds without being pushed out. The constant  $k$  is chosen to be positive with no other constraints; a practical value of 0.4 has been selected for our applications. The process for calculating these parameters and their impact on the trajectory of the tool within the container is visualized in Figure 5. This visual aid helps understand how the spiral's section size and the chosen value of  $k$  influence the movement pattern within the specified constraints.



**Figure 5:** Sampling Rates and the Resulting Trajectories

Mix-poses perform the main tasks in mixing, with the "reso" value representing the resolution of the circle or ellipse. Orbital motion combines two circle motions for more complex mixing tasks. This section can be repeated based on the specified number of rounds. If no rounds value is defined, one completed mix-motion is assumed by default. Otherwise, the poses are linked as many times as rounds specify.

## 5. Integrating new actions into NEEM generation

This section will briefly describe the steps needed to achieve NEEM generation of newly introduced actions within the CRAM framework, utilising the Bullet World simulation. It builds upon the previous work of S. Koralewski et.al. [19] and the there presented CRAM Cloud Logger (CCL), which by now has been dramatically extended and adapted to KnowRob 2.0 [8]. The newly extended CRAM Cloud Logger establishes the necessary connection between the CRAM action designators and maps the different terminologies of KnowRob, SOMA [20] and CRAM to each other. The integration of Virtual Reality NEEMs into CRAM for reasoning purposes has been previously presented by Kazhoyan et.al. [12] and can be seen as a proof of concept for this approach. However, VR NEEMS are very different to the NEEMs recorded from simulation for this paper, which will be further explained in the upcoming section.



## 5.1. The differences between Virtual Reality and CRAM Bullet World NEEMs

In the learning from VR NEEMs approach, presented in "Learning Motion Parameterizations of Mobile Pick and Place Actions from Observing Humans in Virtual Environments" [12], the data is obtained by a human user performing everyday manipulation activities in a Virtual Reality environment. That work presents the pipeline of gathering NEEMs in VR, processing them in KnowRob [8] and generalizing the obtained data to determine parameters for executing the VR-shown action within the Bullet World simulation and the real robot. It also shows that the learned parameters can be generalized across different kitchen environments and used by other robots.

Even though the goal of setting the table is given in advance, during the task execution and subsequent NEEM generation in Virtual Reality, it is impossible to know which high-level action the human user is currently performing. This knowledge is extrapolated based on specific events which take place. [21] [22] E.g. if a contact event between the simulated hand occurs with a cup, followed by the loss of contact between the cup and the table surface by which it was supported, it can be assumed that the human user has picked up the cup. Once the cup has lost contact with the hand but a contact event between the cup and the kitchen island has occurred, it can be assumed that the cup has now been transported to a new location and placed there. In contrast, the NEEMs generated from the CRAM Bullet World are based on action designators and their resolution. Since a designator is recorded in its unresolved and resolved form, this kind of NEEM would allow the comparison of different resolution approaches, which was also a crucial part of the evaluation in the Kazhoyan et al. paper [12].

## 5.2. Setting up the NEEM generating environment

The NEEM logging process works by attaching additional functions to the designator execution function within CRAM, which allows the designator and all of its parameters, both before and after the resolution, to be logged via queries to KnowRob, which then writes the data into an instance of MongoDB. However, it is necessary to link the CRAM designator terminology to the SOMA ontology used by the KnowRob framework so that the NEEM can be queried for information and reasoned about later. This is done by introducing additional resources to KnowRob, such as the environment urdf files and the robotic agent. In addition, owl files are needed to describe the contents of the URDF files semantically and to establish the connection between the robot/environment description and the semantics used within KnowRob. Since KnowRob can utilise "rospack" services, the paths can be given accordingly. This Parameter setting for CRAM Cloud Logger is shown in Listing 2.

Listing 2: Environment Parameter settings for the CRAM Cloud Logger; extract of a setup-logging-function

---

```
(setf *environment-owl* "'package://iai_semantic_maps/owl/kitchen.owl'")
(setf *environment-owl-individual-name* "'http://knowrob.org/kb/
IAI-kitchen.owl#iai_kitchen_room_link'")
(setf *environment-urdf* "'package://iai_kitchen/urdf_obj/kitchen.urdf'")
(setf *environment-urdf-prefix* "'iai_kitchen/'")
(setf *agent-owl* "'package://knowrob/owl/robots/PR2.owl'")
(setf *agent-owl-individual-name* "'http://knowrob.org/kb/PR2.owl#PR2_0'")
(setf *agent-urdf* "'package://knowrob/urdf/pr2.urdf'")
```

---

The utilised objects also have to be mapped to the SOMA ontology, e.g. the BOWL's notion must be mapped to:

<http://www.ease-crc.org/ont/SOMA.owl#Bowl>

This must be done for every new notion of objects or actions accordingly. Since SOMA has been used previously for robotic NEEM generation, many of the necessary objects are already represented in the ontology. If such a representation does not exist, the overarching class can be used instead. E.g. since WHISK as a tool does not yet exist, it can be represented as a DesignedTool instead

which is described as "An item designed to enable some action, in which it will play an instrumental role." Alternatively, the necessary object can be added to the ontology by modifying SOMA directly. This approach would require, however, that the changes be submitted to the official SOMA repository so that everybody can use them. Some CRAM action key-value pairs formats, such as poses, must then be converted into a format that KnowRob supports. This coupling of SOMA and CRAM links CRAM actions and semantic knowledge. The low-level data, e.g. robot joint positions, is directly logged from the ROS tf topic and is being synchronised with the individual performed actions via time stamps.

### 5.3. NEEM generation

After the mapping is complete, the NEEM generation can begin. For this, KnowRob has to be running with an instance of MongoDB, and the executed plan has to be wrapped into start- and stop-episode functions. The demo within the start and stop statements can also be executed multiple times to obtain an NEEM containing multiple task executions. An example query which is generated during runtime then to log actions and their subactions can look like this:

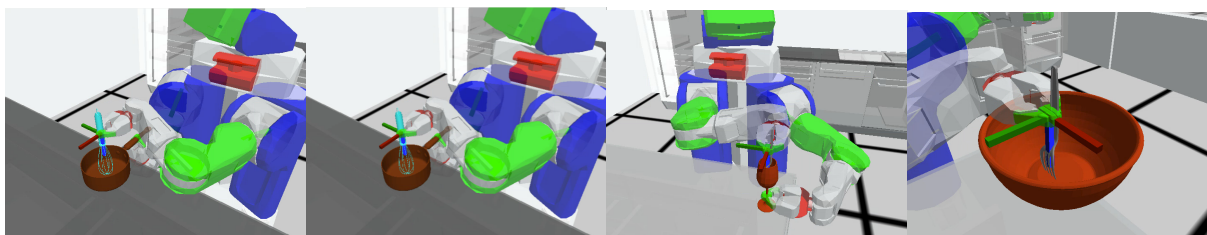
Listing 3: Example of a query which adds a new subaction of type PickingUp.

```
add_subaction_with_task (  
  'http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Action_FKYRWDBL',  
  SubAction,  
  'http://www.ease-crc.org/ont/SOMA.owl#PickingUp')
```

The resulting NEEM can then be loaded into a local instance of KnowRob or into OpenEase to be queried and utilised. Loading NEEMs into a locally running instance of KnowRob allows them to be queried from CRAM, which is the planning and execution system. The queries can be used to acquire the parameters with which the task execution was successful and in which the simulated robot fulfilled the previously found conditions, e.g. the whisk was not dropped and was touching the bowl and mixture during the mixing action. The results of the queries can then be used within the action designator to perform the action.

## 6. Discussion

Our theoretical model provides a structured approach to identifying the core aspects of a task and transforming them into executable steps. To validate this hypothesis, we conducted an experiment using the PR2 Robot, a mobile manipulation platform Willow Garage developed to perform the specified action designator within Bullet World, a rapid-fire simulation. To support our analysis, the experiment was documented through recorded videos and screenshots, accessible online. Figure 6 displays a visual summary of the experiment, which illustrates how the theoretical model is applied in a practical scenario, showcasing the robot's capacity to execute the designated tasks. In the experiment, we



(a) Mixing with Whisk in Pot

(b) Mixing with Whisk in Bowl

(c) Mixing with Fork in Wineglass

(d) Mixing with Fork in Bowl

**Figure 6:** Specific Mixing Task variations performed by PR2 robot.



used four containers and three types of tools in various combinations, thoroughly documenting the process for evaluation. The containers included a large bowl, a saucepan, a smaller round bowl, and a wineglass, while the tools consisted of a whisk, a fork, and a spoon. It is important to note that some combinations, such as a fork with a wineglass and a whisk with a small round bowl, though operational with the action designator, showed limited success due to the spatial constraints within the containers. Overall, the action designator was successful in most scenarios, except when the tool's width was too great for the container's capacity, leading to errors. This highlights the progression from a theoretical model to a versatile action designator capable of handling diverse scenarios with different tools, although limitations from the simulation still persist, such as not fully addressing factors like mixture homogeneity, the viscosities of liquids and substances, and the optimal speed for executing actions.

## 6.1. Potential applications of Bullet World NEEMs

To further evaluate the success of the action, we recorded Narrative-Enabled Episodic Memories (NEEMs) during the experiment. These NEEMs can provide crucial insights into the parameters for successful action execution. We showcase one such recorded NEEM, available [online](#), and discuss the potential insights gained from this and future recordings. So far, we have looked into NEEM generation in VR and in simulation. VR NEEMs have been already used to teach robots new manipulation activities and have their benefits, however NEEMs from simulation can be used to enhance a robot's performance as well but in different ways. Since the Bullet World simulator is a rapid-fire simulator, it is very fast but omits certain visualisations. For example, the robot is teleported to a target location instead of the entire path being simulated and visualised. (Details of the implementation and inner workings can be found in Kazhoyan [13].) This way, time can be saved on trajectory simulation, and the gained speed can be utilised for testing various configurations before execution on the real robot [13]. However, since the simulation does not provide information such as collision checking and physics, which is simulated for a split second to check if items would fall or slip out of the robot's hand, it is enough to estimate if certain actions would be successful. From KnowRob [23] [8] and based on previously collected VR NEEMs [21] [22], we could manually gather the **conditions** which have to be met to achieve a **successful mixing action**. Such conditions could be touching events between the tool and the robot's gripper, the fact that the tool has not been dropped during the action, a touching event between the tool and contents of the bowl during the mixing process must occur over some time or repeated multiple times, to achieve the effect of combining two ingredients. If these conditions are met, the simulated mixing action will likely be successful, even though the Bullet World could not have simulated the physical effect of mixing.

This idea holds much potential for automating NEEM evaluation and selecting the best action parameters based on prior experiences. By obtaining a large number of NEEMs in which different parameters were used during task execution, it is possible to determine the potentially best parameters for an action and also estimate the robot's success during task execution. Another benefit to this approach is that bullet NEEMs require fewer resources to generate compared to VR NEEMs. For the latter, more hardware is needed in the shape of VR Headsets, controllers, base stations, a decent GPU which can run *Unreal Engine*, and a human user who performs the actions in VR in real-time. For Bullet World NEEM generation, a computer with Ubuntu 20.04 and the software stack consisting of CRRM, KnowRob, MongoDB and ROS is needed. An onboard GPU is enough; the experiments can run in a loop overnight to generate the data.

## 7. Future Outlook

Looking ahead, our future research will continue to advance the integration of cognitive robotics into everyday environments with several strategic initiatives:

We plan to investigate the effects of different container shapes on robotic mixing capabilities. By understanding how various shapes influence the mixing process, we can refine our algorithms to handle

a wider range of household tasks, enhancing the robot’s versatility and effectiveness. We will also deepen our use of Narrative-Enabled Episodic Memories (NEEMs) to improve the robot’s learning and decision-making processes. By expanding the NEEM dataset and enhancing the analysis techniques, the robot can learn from past actions and adjust its strategies more effectively. This ongoing learning will be critical for developing robots that adapt to new tasks with minimal human intervention. We will conduct extensive real-world testing to validate and refine our models based on our theoretical and simulated advancements. Testing in actual home environments will help identify practical challenges and user needs, ensuring that our robots can operate effectively in their intended settings. As robots can perform complex tasks independently, we will explore collaborative scenarios where robots work alongside humans or other robots. This research will focus on optimizing task division, improving interaction protocols, and ensuring safety in shared spaces.

By pursuing these roads, we aim to push the boundaries of what cognitive robots can achieve, making them more aware, adaptable, and useful across various settings.

## Acknowledgments

The research reported in this paper has been partially supported by the German Federal Ministry of Education and Research; Project-ID 16DHBKI047 “IntEL4CoRo - Integrated Learning Environment for Cognitive Robotics”, University of Bremen as well as the German Research Foundation DFG, as part of Collaborative Research Center (Sonderforschungsbereich) 1320 “EASE - Everyday Activity Science and Engineering”, University of Bremen (<http://www.ease-crc.org/>). The research was conducted in subproject R04 “Cognition-enabled execution of everyday actions”.

## References

- [1] M. M. Hedblom, M. Pomarlan, R. Porzel, R. Malaka, M. Beetz, Dynamic action selection using image schema-based reasoning for robots, 2021.
- [2] G. Kazhoyan, S. Stelter, F. K. Kenfack, S. Koralewski, M. Beetz, The robot household marathon experiment, IEEE International Conference on Robotics and Automation (ICRA) (2021). URL: <https://arxiv.org/abs/2011.09792>, (Accepted for publication).
- [3] H. Li, D. Xu, An overview of fluids mixing in t-shaped mixers, Theoretical and Applied Mechanics Letters 13 (2023). URL: <https://www.sciencedirect.com/science/article/pii/S2095034923000375>. doi:<https://doi.org/10.1016/j.taml.2023.100466>.
- [4] M. Kümpel, V. Hassouna, J.-P. Töberg, P. Cimiano, M. Beetz, Cut, chop, slice or dice: Parameterising general action plans using knowledge graphs, in: International Conference on Intelligent Robots and Systems (IROS 2024), 2024. Submitted.
- [5] K. Dhanabalachandran, V. Hassouna, M. M. Hedblom, M. Kümpel, N. Leusmann, M. Beetz, Cutting Events: Towards Autonomous Plan Adaption by Robotic Agents through Image-Schematic Event Segmentation, in: Proceedings of the 11th on Knowledge Capture Conference, K-CAP ’21, Association for Computing Machinery, New York, NY, USA, 2021, pp. 25–32. doi:10.1145/3460210.3493585.
- [6] M. Beetz, G. Kazhoyan, D. Vernon, The cram cognitive architecture for robot manipulation in everyday activities, 2023. URL: <https://arxiv.org/pdf/2304.14119.pdf>. arXiv:2304.14119.
- [7] J. Winkler, M. Tenorth, A. K. Bozcuoglu, M. Beetz, Cramm-memories for robots performing everyday manipulation activities, Advances in Cognitive Systems 3 (2014) 47–66. URL: <https://www.semanticscholar.org/paper/CRAMm-Memories-for-Robots-Performing-Everyday-Winkler-Tenorth/9c8c2448a033da67f6bea2a4d89fe5c77cfb3b00>.
- [8] M. Beetz, D. Beßler, A. Haidu, M. Pomarlan, A. K. Bozcuoglu, G. Bartels, Knowrob 2.0 – a 2nd generation knowledge processing framework for cognition-enabled robotic agents, in: In-

- ternational Conference on Robotics and Automation (ICRA), Brisbane, Australia, 2018. URL: <https://ai.uni-bremen.de/papers/beetz18knowrob.pdf>.
- [9] S. Warren, P. Artemiadis, On the control of human-robot bi-manual manipulation, volume 78, Springer Science and Business Media LLC, 2015, pp. 21–32. URL: <https://doi.org/10.1007/s10846-014-0055-4>.
- [10] M. A. Bollini, Following recipes with a cooking robot, 2012. URL: <https://dspace.mit.edu/handle/1721.1/74451>.
- [11] K. Ikeda, H. Masuda, N. Shirasugi, et al., Cfd analysis of effective human motion for whipping heavy cream by hand, volume 75, AIDIC, 2019, pp. 121–126. URL: <https://doi.org/10.3303/CET1975021>. doi:10.3303/CET1975021.
- [12] G. Kazhoyan, A. Hawkin, S. Koralewski, A. Haidu, M. Beetz, Learning motion parameterizations of mobile pick and place actions from observing humans in virtual environments, in: 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2020, pp. 9736–9743. doi:10.1109/IROS45743.2020.9341458.
- [13] G. Kazhoyan, M. Beetz, Executing Underspecified Actions in Real World Based on Online Projection, in: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE Computer Society, Macau, China, 2019, pp. 5156–5163. doi:10.1109/IROS40897.2019.8967867.
- [14] M. Rohrbach, et al., A database for fine grained activity detection of cooking activities, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012. URL: <https://www.mpi-inf.mpg.de/departments/computer-vision-and-machine-learning/research/human-activity-recognition/mpii-cooking-activities-dataset>.
- [15] Wikihow, WikiHow, ??? URL: <https://www.wikihow.com>.
- [16] M. Kümpel, J.-P. Töberg, V. Hassouna, M. Beetz, P. Cimiano, Towards a knowledge engineering methodology for flexible robot manipulation in everyday tasks, in: Workshop on Actionable Knowledge Representation and Reasoning for Robots (*AKR<sup>3</sup>*) at Extended Semantic Web Conference (ESWC 2024), 2024. In press.
- [17] How Things Fly, Kepler’s laws of orbital motion, no date. URL: <https://howthingsfly.si.edu/flight-dynamics/kepler-3s-laws-orbital-motion>, homepage for How Things Fly. [Viewed 21 March 2024].
- [18] J. R. Flanagan, M. C. Bowman, R. S. Johansson, Control strategies in object manipulation tasks, *Current Opinion in Neurobiology* 16 (2006) 650–659. doi:10.1016/j.conb.2006.10.005.
- [19] S. Koralewski, G. Kazhoyan, M. Beetz, Self-specialization of general robot plans based on experience, *IEEE Robotics and Automation Letters* 4 (2019) 3766–3773. doi:10.1109/LRA.2019.2928771.
- [20] D. Beßler, R. Porzel, M. Pomarlan, A. Vyas, S. Höffner, M. Beetz, R. Malaka, J. Bateman, Foundations of the socio-physical model of activities (soma) for autonomous robotic agents (2020). URL: <https://doi.org/10.48550/arXiv.2011.11972>. arXiv:2011.11972, submitted on 24 Nov 2020.
- [21] A. Haidu, M. Beetz, Automated models of human everyday activity based on game and virtual reality technology, in: 2019 International Conference on Robotics and Automation (ICRA), 2019, pp. 2606–2612. doi:10.1109/ICRA.2019.8793859.
- [22] A. Haidu, M. Beetz, Action recognition and interpretation from virtual demonstrations, in: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2016, pp. 2833–2838. doi:10.1109/IROS.2016.7759439.
- [23] M. Tenorth, M. Beetz, KNOWROB - Knowledge Processing for Autonomous Personal Robots, in: 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, St. Louis, MO, USA, 2009, pp. 4261–4266. doi:10.1109/IROS.2009.5354602.

## A. Online Resources

All referenced online sources are available via

- MongoDB: <https://www.mongodb.com/>

- SOMA - Socio-physical Model of Activities: <https://ease-crc.github.io/soma/>
- OpenEase: <http://www.open-ease.org/>
- NEEM Handbook: <https://ease-crc.github.io/soma/owl/1.1.0/NEEM-Handbook.pdf>
- Recorded videos: <https://sunava.github.io/cram-robot-actions/mixing.html>
- Recorded NEEMs: [https://data.open-ease.org/QA?neem\\_id=65fc2047ac7bd8f0875c795a](https://data.open-ease.org/QA?neem_id=65fc2047ac7bd8f0875c795a)
- Unreal Engine: <https://www.unrealengine.com/>
- ROS TF library: <http://wiki.ros.org/tf>