DOI: 10.2195/lj_proc_enke_en_202211_02
URN: urn:nbn:de:0009-14-55862

Learning from Demonstration in Material Handling Processes

Lernen aus Demonstrationen in Handhabungsprozessen

***Constantin Enke*[1]**
***Jonathan Auberle*[2]**

[1]*Institute for Material Handling and Logistics, Karlsruhe Institute of Technology, Karlsruhe, Germany*
[2]*Servus Intralogistics GmbH, Dornbirn, Austria*

Processes in material handling must be flexible and easily adaptable. It is simple for a human to learn to grasp a box from a shelf. To teach a robot to do the same requires programming skills and therefore skilled personnel. Because of this the *Learning from Demonstration* (LfD) approach is gaining importance in recent years. A robot learns from a human demonstrating a task and then reproduces it in new situations. In the area of material handling many situations could benefit from the use of robots, but the implementation often fails because of complex programming or the lack of flexibility of the automated solutions. Therefore, a framework is presented, that is tailored to these specific requirements. The *5+5 Steps of the Material Handling Loop* propose that most tasks in material handling can be segmented into simpler rules. Each of these tasks consist of picking up an object from a source, moving it to a sink and placing it down again. The flexibility of this approach was investigated in two experimental series. While there are still some shortcomings and open issues, it is shown, that this framework enables adaptive and flexible applications for LfD in material handling processes.

[Keywords: Material Handling Processes, Learning from Demonstrations, Flexibility, Robotics]

Prozesse in der Materialhandhabung müssen flexibel und leicht anpassbar sein. Ein Mensch kann einfach lernen, eine Kiste aus einem Regal zu greifen. Einem Roboter dasselbe beizubringen, erfordert Programmierkenntnisse und daher Fachpersonal. Aus diesem Grund gewinnt der Ansatz des Lernens von Demonstration (LfD) in letzter Zeit an Bedeutung. Der Roboter lernt von einem Menschen, der die Aufgabe vorführt, und reproduziert sie dann in neuen Situationen. Im Bereich der Materialhandhabung gibt es viele Situationen, die vom Einsatz von Robotern profitieren könnten, aber die Umsetzung scheitert oft an komplexer Programmierung oder mangelnder Flexibilität der automatisierten Lösungen. Daher wird ein Framework vorgestellt, das auf diese spezifischen Anforderungen zugeschnitten ist. Die 5+5 Schritte des Material-Handling-Loops basieren auf der Annahme, dass die meisten Aufgaben in der Handhabung in einfachere Regeln unterteilt werden können. Für jede Aufgabe muss ein Objekt in einer Quelle aufgenommen werden, zu einer Senke transportiert und dort wieder abgelegt werden. Die Flexibilität dieses Ansatzes wurde in zwei Versuchsreihen untersucht. Obwohl es noch Verbesserungspotentiale gibt, konnte gezeigt werden, dass dieses Framework adaptive und flexible Anwendungen für das LfD in Materialhandhabungsprozessen ermöglicht.

1 INTRODUCTION

Material handling systems must be flexible to be profitable for pick and place tasks in the logistics and intralogistics context. They have to deal with environmental changes, be fast in their task execution, and easily adaptable to new situations and tasks [1]. Because of labor shortages, humans must work more efficiently in material handling processes being supported by robots. Especially in intralogistics where many tasks are not automated yet.

For a human, it is easy to pick up a box from a shelf and place it on a trail or to grasp a certain object from one crate and place it in another one. Teaching the same task to a robot is difficult and always needs skilled personnel. Being able to teach robots new tasks without the necessity for programming would be the optimal solution to increase the degree of automation in intralogistics. Because of this, *Learning from Demonstration* (LfD) has become increasingly more popular over the last few years. Humans demonstrate a task to a robot and the robot reproduces it and adapts the learned skills to new configurations of the environment. Despite many approaches for LfD, there is not yet one that deals with the special challenges of material handling tasks.

Most of the existing approaches in LfD try to learn general behavior [2]. If it is possible to break down material handling tasks into smaller segments that are similar in each repetition of the task, it is possible to create a framework that is not necessary to learn general behavior. For this, a set of material handling tasks was investigated more closely (see Figure 1). First a simple pickup task, then a

DOI: 10.2195/lj_proc_enke_en_202211_02
URN: urn:nbn:de:0009-14-55862

pick from a shelf by lifting a box and pulling it from the shelf, and third, opening a box and retrieving the content.

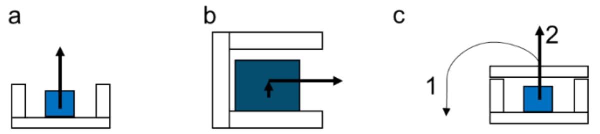

Figure 1: A set of pick-and-place tasks in material handling that were investigated with different start positions to find similarities in the movements for a) picking an object from a box, b) removing an object from a shelf and c) first opening a box then retrieving the content.

For each of these three tasks, twenty random starting positions were chosen in the robot's workspace. The robot was then manually guided by a human to perform the task. The resulting trajectories were recorded and analyzed to find patterns. The resulting tractories of a pick from a shelf (Figure 1 b), divided into x, y, and z axes, are shown in Figure 2 as continuous lines in different colors.

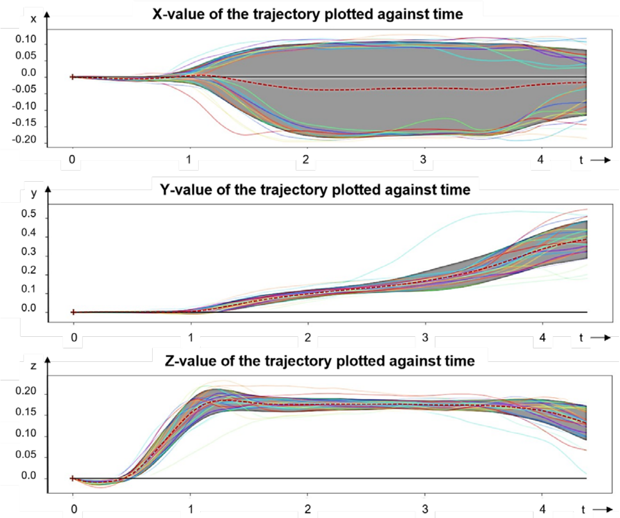

Figure 2: Investigation of the normal distribution of the trajectories after equating the start position.

To normalize the trajectories in their grasp point ($t = 0$), the x, y, and z values of the grasp position were subtracted from each point of the recorded trajectory. So, the trajectory always starts in the position (0,0,0). In Figure 2 the resulting standard deviation is visualized as the grey area, while the average value is shown as a dotted red line.

The standard deviation around the time of the grasp is smaller than later in the task. Analysis of the other two tasks lead to the same outcome. Based on this it can be assumed that the movement near the grasp point is similar in every instance of the task execution and thus can be segmented as a pick-up movement from the rest of the task. In this work a framework is presented to apply LfD in material handling processes based on this hypothesis. By breaking down tasks into segments an easy adaptable learning and a flexible execution becomes possible.

In Section 2 previous works related to this paper are introduced. In Section 3 a framework for LfD in material handling processes is presented. To show the validity of the created framework, two experimental series investigate its flexibility in Section 4. Finally, in Section 5, the work is summarized and an outlook for future work of LfD in material handling processes is given.

2 Related Work

2.1 Bottom-Up Approach to describe Material-Handling Processes

The concept of breaking down intralogistics processes into segments stems from Furmans and Gue [3] who present a bottom-up approach. According to them, a vast majority of intralogistics processes can be described by elementary functions which are combined to complex processes. Functions are understood as the basic elements of material handling and can be physical (influencing the real world) or cyber functions (describing the interaction between components). Among the seven physical functions Store, Move, Transfer, Pick, Place, Unitize, and Separate, three functions are relevant for this work in more detail:

- *Move*: The move function changes the position or orientation of an object without changing the material carrier or the environment. This includes both the movement of objects and the movement of the modules themselves. For example, a driverless transport vehicle can use the Move function to move a crate from the pickup location to the storage location as well as to move itself along this route.
- *Pick*: The pick function refers to the selection and removal of an object from a load carrier. For example, a screw can be picked up from a crate. In contrast to the Move function, the object was in a load carrier before it was picked up. In addition, other objects can remain there after picking.
- *Place*: With the place function, an object is placed in a load carrier. This function is the inverse of the pick function.

2.2 Approaches to Learn from Human Demonstrations

In the field of LfD, three types of teaching are distinguished [4]. *Kinesthetic Teaching* allows the user to physically move the robot by hand to teach new tasks. By pressing buttons on intermediate points of the trajectory to be reproduced, trajectories can be specified with any degree of precision. The opening and closing of the grippers are also performed by the control panel attached to the robot. A robot must be specially equipped for this kind of LfD. This differs from *Teleoperation* where the robot is guided by a

DOI: 10.2195/lj_proc_enke_en_202211_02
URN: urn:nbn:de:0009-14-55862

controller, which is not directly attached to the robot. This allows LfD for industrial robots, where no human is allowed in the working area of the robot during the movement. In both these approaches, the robot can learn motions by observing his position and joint ankles. In the third approach of *Passive Observation*, the teacher is observed executing the tasks through sensors and the system derives the necessary motion and transfers it to the joint movements of the robot.

The learning outcome is divided into three categories: *Policy*, *Cost/Reward*, and *Plans*. [4]. Policies describe the direct mapping of input to output data based on the demonstrations. Cost/reward approaches assume that a human in the demonstration is trying to optimize an unknown cost function that must be determined. Plans provide the highest level of abstraction and represent a sequence of primitive actions that can be either sequential or hierarchical.

2.3 Dynamic Movement Primitives

Dynamic Movement Primitives were first introduced by Ijspeert et al. [5]. This approach describes each degree of freedom of a robot as a linear spring-damper system stimulated by a nonlinear external force. DMPs are a deterministic, implicit time-dependent motion representation that allows start, goal, and velocity adjustment. The demonstrated trajectories are recorded by storing timestamp and position in a certain frequency. Using a canonical system, DMPs are independent of time by replacing the time variable of the trajectory with a phase variable s that runs from 1 at the starting point to 0 at the destination point. This leads to the equation

$$\dot{\tau v} = K(g-x) - Dv + f(s)$$

with K, D as variables of the spring-damper system and $f(s)$ the non-linear term, which is trained from by the demonstrated trajectories. Start x_0 and target g can be adjusted in the transformation system. By inserting the phase variable s the velocity and the trajectory for this degree of freedom can be determined (Figure 3).

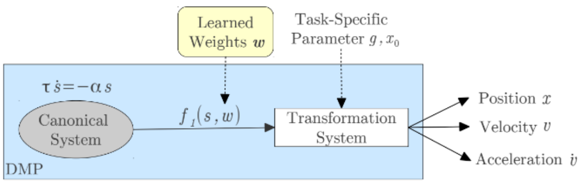

Figure 3: Summary of the one-dimensional DMPs with a canonical syst stem, which specifies the non-linear part and thus can react to new task parameters [8].

In [6] the concept of a library to store sets of movement primitives. This leads to a framework is introduced. During execution, the appropriate movement is selected from the library depending on the observation of the situation (see Figure 4). In [7] changes to the scalability and rotation of DMPs are made, that lead to better adoptions to new situations.

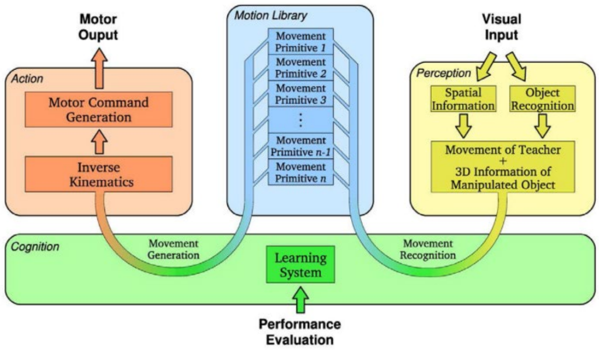

Figure 4.: Schematic representation of the DMP-based control of a movement using a library of movement primitives [8].

3 LfD in Logistic Processes

As stated in chapter 1 material handling tasks often have a modular structure. This can be used to improve LfD since a shorter horizon of tasks leads to an improvement of the learning and reproduction [8]. In this work, the framework *5+5 Steps of the Material Handling Loop* is introduced. To use this framework, the system must be provided with information about the position and orientation of source and sink from external devices, such as sensors. Only so the flexibility made possible by the framework can be used. Figure 5 shows the concept build from several trajectory segments, which are needed to execute material handling tasks of various difficulties. All tasks aim to move an object from a source to a sink. To do so it can be necessary to prepare or postprocess the source or sink for example by opening or closing a box. Also, the possibility to move between source and sink without damaging the environment is needed. Flexibility is introduced through the reference coordinate systems (CS) source, sink and world, relative to which the trajectories are to be executed. In the following chapters this framework is built up gradually and the different components are explained in more detail.

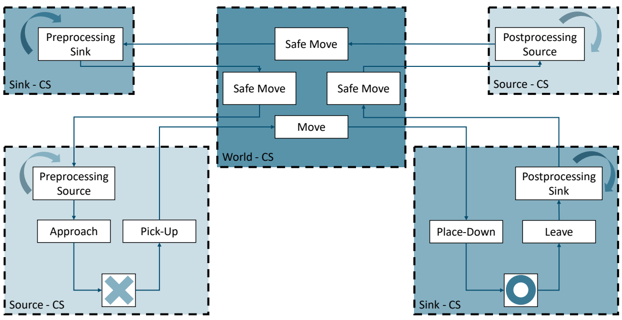

Figure 5.: Conceptual description of the 5+5 Steps of the Material Handling Loop with trajectory segments (white boxes), the grasp (x) and the release (o) point and the reference coordinate systems (CS) source, sink and world, relative to whom the trajectories are executed.

DOI: 10.2195/lj_proc_enke_en_202211_02
URN: urn:nbn:de:0009-14-55862

3.1 Structure of Logistics Tasks

The elementary functions of each task are, as defined in the Material Handling Framework [3], *Move*, *Pick* and *Place*. These functions are the higher-level segments of the task, represented using capital letter Π. The pick function describes the pick-up movement from its source written as Π_{pick}. The place function Π_{place} represents the placing of the object in a sink. The move function Π_{move} represents the movement of the robot, both loaded and unloaded, through a workspace. To be reproduced by the robot, these functions are divided further into rules on a lower level. The notation used in this work is following a commonly used mathematical representation in the field of LfD [4].

Table 1: List of notations used in this work

Notation	Description
π_i	Rule for performing action i always consists of a start state A an end state B and a trajectory σ_i: $\pi = A \overset{\sigma_i}{\rightarrow} B$
σ_i	Trajectory, necessary for the transition from state A to state B
$O/I(./o_j)$	Source (O)/sink (I), either empty or with free capacity (.) or containing the object to be grabbed (o_j)
$R(./o_j)$	State of the robot with empty gripper or loaded with object o_j
A/B	State before/after performing a trajectory, consisting of the states of all participants in the system. For example: $A = O(o_j) \cup I(.) \cup R(.)$ describes the state where the object is in the source while the sink and robot have the capacity to hold the object

3.2 Segmentation of the Motion Sequences

3.2.1 5 Essential Steps of Logistic Processes

For a material handling task, a robot must execute *move* to the source, *pick* the object, *move* the object to the sink, *place* it down again and *move* to the next task. The pick function Π_{pick} consists of two rules. Using the example of a grasp from a shelf by a two-finger gripper, a trajectory is required that approaches the correct shelf and ends in a way that the object lies between the fingers. This movement rule is defined as $\pi_{approach}$. In the next step, the gripper closes so the object is loaded and a trajectory is executed to leave the source, defined by π_{pickup}. Π_{place} correspondingly consists of two steps as well – $\pi_{placedown}$ and π_{leave}.

The corresponding rule π_i that applies to each step is represented as a transition from the start state A to the target state B by using the corresponding trajectory σ_i

$$\pi_i = A \overset{\sigma_i}{\rightarrow} B.$$

For the sake of clarity, the state of source or sink are from here on only displayed if a change in its state occurs, otherwise they are left out. The five essential steps are defined as follows:

- *Approach*: The robot enters the source that contains the object to be grabbed. After the rule execution, the empty robot $R(.)$ is therefore in the source O, which contains the object to be grasped o_j:
$$\pi_{approach}: O(o_j) \cup R(.) \overset{\sigma_{ap}}{\longrightarrow} O\left(o_j, R(.)\right)$$
- *Pick-up*: The robot picks up the object from the source. The rule describes the transition to an empty source and a robot holding the object outside the source:
$$\pi_{\mathrm{pickup}}: O\left(\mathrm{o_j}, \mathrm{R}(.)\right) \overset{\sigma_{up}}{\longrightarrow} \mathrm{O}(.) \cup \mathrm{R}(\mathrm{o_j})$$
- *Move*: The robot moves together with object o_j from the end of the pickup motion to the beginning of the deposit motion, following the boundary conditions of the workspace:
$$\pi_{move} = \sigma_{auf,end} \xrightarrow{\sigma_{move}} \sigma_{place,start}$$
- *Place-down*: The robot enters the sink I with the grasped object o_j. In the sink is enough space for the object to be placed:
$$\pi_{placedown}: I(.) \cup R(o_j) \xrightarrow{\sigma_{down}} I\left(R(o_j)\right)$$
- *Move-away*: The robot places the object in the sink and leaves it again. After the rule is executed, the sink contains the object, and the robot is in a free position:
$$\pi_{leave}: I\left(R(o_j)\right) \xrightarrow{\sigma_{leave}} I(o_j) \cup R(.)$$

For the transition between the different steps of the process, the corresponding preconditions must always be fulfilled. For example, for the start-up it is necessary that the object to be grasp is in the source and that the robot has the capacity to pick up the object at the same time. After the completion of each step, it is necessary to check whether the corresponding final state has been reached.

3.2.2 5 Additional Steps for Complete Coverage

To ensure a continuous process by stringing together several tasks, i.e., the *move* from sink to source, it is necessary to define a *safe transition*. A safe transition $\pi_{safemove}$ describes the movement of the unloaded robot from the end position of the last action σ_{i-1} to the next action σ_i without posing a danger to the workspace. The most direct solution

DOI: 10.2195/lj_proc_enke_en_202211_02
URN: urn:nbn:de:0009-14-55862

to that is the introduction of a safe movement height, in which the robot can reach all points without collision.

To deal with more complex tasks, a pre- and post-processing of the source and sink are introduced. This covers actions like unlocking a shelf/locking it after the task or opening and closing a box to retrieve an object. The sequence is a preprocessing of the sink $\pi_{pre,sink}$, followed by the preparation of the source $\pi_{pre,source}$. Then the task is executed as presented in Chapter 3.2.1. Finally, first the sink is postprocessed $\pi_{post,source}$ and then the source $\pi_{pre,source}$. This order is reducing the necessary movement between preprocessing of the source and the pick as well as between placing and postprocessing of the sink.

3.2.3 Introduction of Separate Coordinate Systems

Changes in position of source and sink are adopted by adjusting Π_{pick} and Π_{place} to the new positions. Dealing with changes in orientation on the other hand can be difficult for LfD algorithms [7]. Because of this, relative coordinate systems (CS) were introduced for source, sink and world. All rules related to the source ($\pi_{approach}$,π_{pickup}, $\pi_{pre,source}$, $\pi_{post,source}$) are to be executed in the source CS. Same is true for the sink. For example, to approach a box from the correct angle, the trajectory is generated as if there was no rotation and is then transformed into the relative CS. In this way, the complex setup of learning grasps for different orientations [7] can be bypassed in the use case of material handling tasks. Movements that are independent from source and sink (π_{move}, $\pi_{safemove}$) are executed in the world CS and therefore must not be transformed.

3.3 Application

LfD is split into a learning/teach-in phase and a reproduction phase. The learning phase generates rules that need to be stored. In the reproduction phase, the situation must be observed and the right rules for this situation must be applied.

In the learning phase the trajectory necessary to execute the task must be recorded, as well as grasp and release positions. Additionally, an external observation of the task is necessary to retrieve the necessary information on source, sink, their orientation, and the type of task that is demonstrated. As shown in Figure 6 trajectory and task understanding must be merged to create corresponding rules. For example, human input can be used to define, when the $\pi_{approach}$ movement is started. This section ends with a grasp action. The trajectory to train $\pi_{approach}$ is σ_{ap} with starting position $\sigma_{ap}(s=1)$ being the position defined by human input and goal position $\sigma_{ap}(s \rightarrow 0)$ being the position of the grasp. With start, goal, and the trajectory a DMP for each degree of freedom can be trained and the associated weights can be stored. The rule $\pi_{approach}$ can then be chained with several $\pi_{pre,i}$ with i being the steps of the preparation and a π_{up} rule to the pick function $\Pi_{pick} = \pi_{approach} + \sum \pi_{pre,i} + \pi_{up}$. This function is then stored in a library in combination with a task description and the original start and goal position. By storing these primitive actions, it is possible to later reproduce different situations, because a pick from a box, for example, differs from a pick from a shelf, so two different basic functions are necessary.

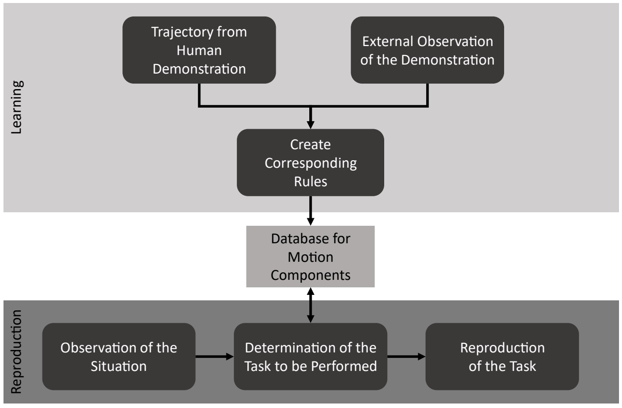

Figure 6.: Sequence of the two-stage process of learning and reproduction with the exchange of the acquired rules via a database that can be accessed from both subprocesses.

For the reproduction first an observation of the situation is necessary. The task must be recognized by assigning task, source, and sink. Alternatively, a request to transport from a defined position of a source to the specified position sink can be made by a higher-level system. The most suitable motion elements for this task are selected from the stored motion elements. If no demonstration is known for the desired combination of source, sink and object, a new teach-in process is triggered. If a corresponding entry is found, the material handling loop is executed sequentially, adapted to the new start and target positions and orientations.

4 Experiments and Evaluations

The presented framework to learn material handling tasks from demonstration is evaluated to see if tasks with different levels of complexity can be learned and flexible reproduced is possible.

4.1 Setup

The presented framework is evaluated on a teleoperated robot at the Institute for Material Handling and Logistics (IFL) at the Karlsruhe Institute of Technology (KIT). The robot is using a virtual reality system to teleoperate the robot. The control of the VR system is used to transfer the movement of the human directly to the robot. The collaborative robot *Franka Emika Panda* is used to execute the movement. To observe the environment a *Kinect* camera is installed in a way, so the complete workspace of the robot

DOI: 10.2195/lj_proc_enke_en_202211_02
URN: urn:nbn:de:0009-14-55862

can be observed. The source and sink are attached with *Aruco*-Markers. *Aruco* markers are codes that contain a number. Additionally, the position and orientation relative to the camera can be derived from the captured images. This is a straightforward way to determine position and orientation of source and sink. This setup is shown in Figure 7.

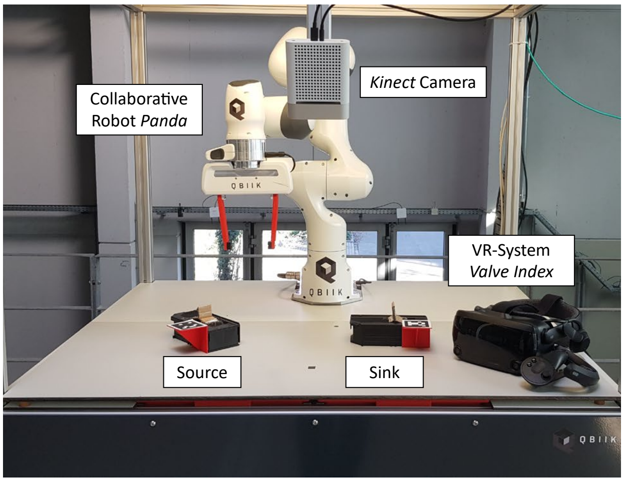

Figure 7.: Setup of the experimental environment with the collaborative robot Panda, the depth camera Azure Kinect, and the teleoperation system with the Valve Index VR glasses

The framework of *5+5 Steps of the Material Handling Loop* can be implemented with various LfD approaches. For this setup, Dynamic Movement Primitives (DMPs) were used since they offer a clear and compact representation of trajectories from a single demonstration with a simple adjustment of the start and goal position while maintaining the shape of the movement.

For demonstrating new tasks, the teleoperation system of the presented setup is used. A human teacher can guide the robot with a handheld controller in real time. Each movement of the hand is directly executed by the robot.

4.2 Experimental Setup

Goal of the experimental series is to show qualitatively that using the introduced framework of *5+5 Steps of the Material Handling Loop* enables LfD in material handling tasks. Also, flexibility towards changes and fast learning are to be investigated and compared to other approaches. The setup of tasks is composed of three different difficulty levels (see Figure 8). Level 1 is a simple pick-and-place task of a block from one crate into another. Level 2 simulates a pick from a shelf, where a block must be lifted first and then pulled out from the shelf. Level 3 is a task with multiple steps, where a box must be opened, a block retrieved from the box and placed in another on. After moving the block, both boxes must be closed again.

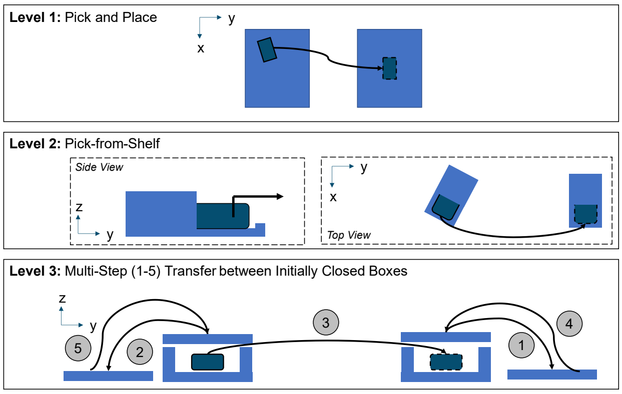

Figure 8.: Scenarios to be evaluated with various levels of difficulty to represent various logistics tasks.

In the first experimental series the proposed framework in the presented setup is tested on the three different levels of complexity by teaching the task one time and trying to reproduce in different configurations in the workspace of the robot. Following the Design of Experiment approach [9], possible configurations of each level were defined by spreading source and sink positions (see Figure 9). Position 1 is the initial source and sink combination used to teach the task, 2 and 3 are in the same position, but the orientation is rotated by $\pm 45°$. Position 4 is the same orientation as position 1 but shifted 10 cm in positive y-direction. The same adaptions were applied to the sink positions. For each configuration it was evaluated if the reproduction was successful to investigate the flexibility of the proposed implementation using the *5+5 Steps of the Material Handling Loop*.

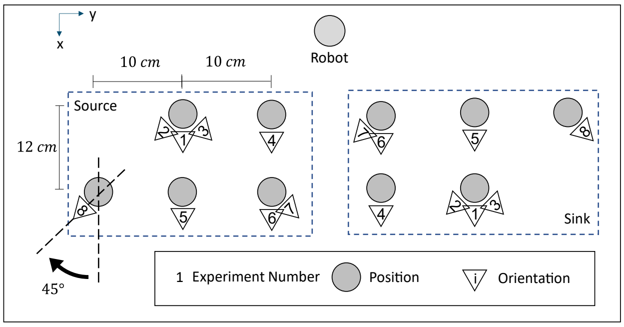

Figure 9.: Experimental setup for evaluating the adaptability of the approach by spreading source and sink positions and rotating them.

In the second experimental series the flexibility of the approach coming from the segmentation of the trajectory is compared to DMPs without this segmentation. For this purpose, the task with difficulty level 2 – *Pick-from-Shelf* was selected. The configuration is shown in Figure 10.

DOI: 10.2195/lj_proc_enke_en_202211_02
URN: urn:nbn:de:0009-14-55862

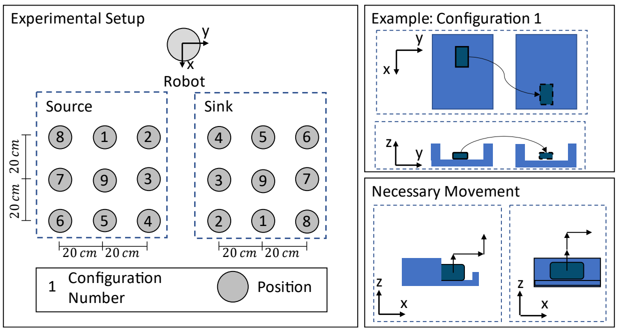

Figure 10.: Setup for the second experimental series to compare the presented approach with unsegmented DMPs performing the described task with start position in source and goal position in sink for each configuration number.

4.3 Learn and Reproduce Different Levels of Complexity

In the first experimental series, the flexibility of the presented approach is examined. The pick-and-place task at difficulty level 1 was be performed in all configurations without any failure. Failure is here defined as not successfully completing the task or colliding with the environment. At level 2, reproduction was possible in six out of eight cases without failure. Configuration 4 was not possible because the boundary conditions of the robot were violated, and in configuration 8 the position of the source was estimated incorrectly by the sensors which lead to a collision with the environment. Level 3 is a complex task. The necessary adaptation of the trajectory from the teaching setup in configuration 1 to the rotated setup in configuration 3 is shown in Figure 11. Here the reproduction was partly successful in 6 out of the 8 configurations. Partly in this case means that all steps have been successful except for the closing of the lids of the boxes (Figure 8, Level 3, Step 4 and 5). For this, a high precision of the reproduction was necessary, which could not be achieved by the setup. The setup also failed in configurations 2 and 8, the positions of the source and sink were also incorrectly determined, and a collision occurred.

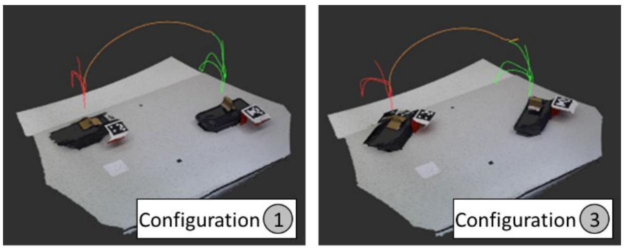

Figure 11.: Visualization of the adapted trajectories from the demonstrated configuration (left) to the 45° rotated configuration (right) with Π_{pick} in red, Π_{move} in orange and Π_{place} in green.

The first experimental series showed a high flexibility of the learned movements to deal with changes in the workspace. In total, 75% of the tests were conducted successful, i.e., the object was moved from the source to the sink. The incorrect 25% can be attributed to hardware restrictions in the robot kinematics and to inaccuracies in image processing to determine positions of source and sink. The inaccuracies of the reproduction are partly explained through the experimental setup, as the lids did not have a rigid structure but were flexible. The topic of precision must be investigated further in the future.

4.4 Comparison with other Approaches

After showing the general flexibility, the influence of the segmentation to this complexity is now investigated. To do so, the approach is compared to the use of DMPs without segmentation. This means the complete trajectory for a task is recorded and transformed into a single set of DMPs. A change of the orientation is not possible here, because the used implementation of the DMPs cannot handle it. For the experimental series, only changes in the position according to Figure 10 were made. The success of the reproductions of the two approaches was compared. The demonstration of the task was done in configuration 1.

Table 2.: .: Success (✔) or failure (✘) of the reproduction using unsegmented and segmented (this approach) DMPs.

	Configuration Number								
	1	2	3	4	5	6	7	8	9
Unsegmented DMPs	✔	✘	✘	✘	✘	✘	✘	✘	✔
This approach	✔	✘	✔	✘	✔	✔	✔	✔	✔

The two times the presented approach failed the reproduction was due to restrictions in the workspace of the robot as already described in the previous chapter. The reproduction using unsegmented DMPs was not successful in seven out of nine cases and a collision with the environment occurred. DMPs learn a specific movement, and reproduction it by scaling it to adopt new start and goal positions. This scaling leads to a distortion of the trajectory. If the new source is to the left (negative y-direction) of the demonstrated configuration 1, the trajectory will be scaled in a way that a collision with the right wall of the source will occur. If the source is moved in positive x-direction to configuration 1, the object will not be pulled out of the source enough and the collision will occur with the top of the source. Here the advantage of the segmentation in this approach can be seen. By segmenting the trajectory, only the pick and place movement is adapted to the source or sink, not the complete trajectory. In addition, rotations are made possible by using this segmentation, since the generated Trajectory can be adjusted in the pick and place segments. For future experiments, this approach must be compared to different LfD approaches that are able to implement rotations. Still, with this experimental series it was shown that by segmenting the trajectory a higher flexibility could be achieved then without it.

5 CONCLUSION AND DISCUSSION

To increase flexibility of material handling tasks, this paper investigated the possibility of LfD in material handling processes. A framework was introduced, that enables a LfD system to react to changes in the position and orientation of source and sink by segmenting the task into a set of rules. The flexibility was shown for the execution of tasks with various levels of difficulty. The approach was also compared to an unsegmented LfD approach. For the conditions investigated, it was shown that the segmentation has a positive effect on flexibility.

Some of the disadvantages that have emerged in the experiments can be attributed to hardware restrictions like the limited workspace of the robot and the technology used to determine position and orientation of source and sink. Another shortcoming is that the presented approach requires a deeper understanding of the task to perform the segmentation of the demonstrated trajectory. This currently means a human must provide this input. It must be investigated in the future if an automatic segmentation is possible. Overall, the approach was only examined qualitatively. In the future, quantitative studies should be conducted to obtain a more accurate picture of the advantages and restrictions of the proposed approach. Additional material handling scenarios must be investigated, to prove the validity of the approach for these scenarios as well. Also, a comparison with other approaches in LfD is necessary, since in this work only the DMPs were considered.

In summary, the framework of the *5+5 Steps of the Material Handling Loop* makes it possible to learn previously unknown material handling tasks and reproduce them in a changing environment, but there are still some points that need to be investigated in the future. By introducing LfD into material handling processes, humans can easily adjust the robots without programming and thus the processes can become more flexible.

LITERATURE

[1] K. Furmans, Z. Seibold, and A. Trenkle, “Future Technologies in Intralogistics and Material Handling,” in *Operations, Logistics and Supply Chain Management*, H. Zijm, M. Klumpp, A. Regattieri, and S. Heragu, Eds. Cham: Springer International Publishing, 2019, pp. 545–574. doi: 10.1007/978-3-319-92447-2_24.

[2] T. Osa, J. Pajarinen, G. Neumann, J. A. Bagnell, P. Abbeel, and J. Peters, “An Algorithmic Perspective on Imitation Learning,” *FNT in Robotics*, vol. 7, no. 1–2, pp. 1–179, 2018, doi: 10.1561/2300000053.

[3] K. Furmans and K. R. Gue, “A Framework for Modeling Material Handling with Decentralized Control,” *15th IMHRC Proceedings (Savannah, Georgia. USA – 2018)*, p. 8, 2018.

[4] H. Ravichandar, A. S. Polydoros, S. Chernova, and A. Billard, “Recent Advances in Robot Learning from Demonstration,” *Annu. Rev. Control Robot. Auton. Syst.*, vol. 3, no. 1, pp. 297–330, May 2020, doi: 10.1146/annurev-control-100819-063206.

[5] A. J. Ijspeert, J. Nakanishi, and S. Schaal, “Movement imitation with nonlinear dynamical systems in humanoid robots,” in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, May 2002, vol. 2, pp. 1398–1403 vol.2. doi: 10.1109/ROBOT.2002.1014739.

[6] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal, “Learning and generalization of motor skills by learning from demonstration,” in *2009 IEEE International Conference on Robotics and Automation*, May 2009, pp. 763–768. doi: 10.1109/ROBOT.2009.5152385.

[7] M. Ginesi, N. Sansonetto, and P. Fiorini, “Overcoming some drawbacks of Dynamic Movement Primitives,” *Robotics and Autonomous Systems*, vol. 144, p. 103844, Oct. 2021, doi: 10.1016/j.robot.2021.103844.

[8] O. Kroemer, S. Niekum, and G. Konidaris, “A Review of Robot Learning for Manipulation: Challenges, Representations, and Algorithms,” *arXiv:1907.03146 [cs]*, Nov. 2020, Accessed: Sep. 05, 2021. [Online]. Available: http://arxiv.org/abs/1907.03146

[9] K. Siebertz, D. van Bebber, and T. Hochkirchen, *Statistische Versuchsplanung*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2017. doi: 10.1007/978-3-662-55743-3.

M.Sc. Constantin Enke, Institute for Material Handling and Logistics (IFL), Karlsruhe Institute of Technology (KIT).

M.Sc. Jonathan Auberle, Servus Intralogistics GmbH, Dornbirn, Austria

Adress: Institute for Material Handling and Logistics, Karlsruhe Institute of Technology, Gotthard-Franz-Str. 8, 76131 Karlsruhe, Germany, Phone: +49 721 608-48632, E-Mail: constantin.enke@kit.edu