J Intell Robot Syst (2017) 85:3–25
DOI 10.1007/s10846-016-0383-7

Task Oriented Control of a Humanoid Robot Through the Implementation of a Cognitive Architecture

Daniel Hernández García · Concepción A. Monje · Carlos Balaguer

Received: 15 July 2015 / Accepted: 22 May 2016 / Published online: 3 June 2016

Abstract This work presents a novel approach on task oriented control of a humanoid robot through the implementation of a cognitive architecture. The architecture developed here provides humanoid robots with systems that allow them to continuously learn new skills, adapt these skills to new contexts and robustly reproduce new behaviours in dynamical environments. This architecture can be thought of as a first stepping stone upon which to incrementally build more complex cognitive processes, providing this way a minimum degree of intelligence for the humanoid robot. Several experiments are conducted to prove the validity of the system and to test the operation of the architecture.

Keywords Learning by demonstration · Motion adaptation · Human-robot interaction · Cognitive models

1 Introduction

A major goal in robotics research is to develop human-like robotic systems capable of interacting and collaborating with humans. Humanoid robots must carry out any number of tasks which their human operators could reasonably expect from them during the normal development of a typical working day. Humanoids must be provided with an architecture that allows them to continuously learn new skills, represent their skill knowledge, and adapt their existing skills to new contexts, as well as to robustly reproduce new behaviours in a dynamic environment in order to cope with working in continuously changing environments and performing a huge variety of tasks.

In our context a skill is defined as a motor trajectory motion learned by the agent, an acquired ability for the execution of a task. A robot skill is a complex action movement, reproducible when appropriate, and generalized to different contexts. Learning systems are required to acquire skills and develop task knowledge of how to act. Algorithms for learning and extracting important features of task actions are fundamental in order to build intelligent behaviours. To learn the skills motion, a time independent model of the motion dynamics is estimated through a set of first order non-linear multivariate dynamical systems. We employ *SEDS* algorithm [23] to learn a global dynamical estimate of the motions through a set of first order non-linear multivariate dynamical systems in a statistical approach.

Despite the clear advantages of learning approaches, it would still be impractical for the human operator to teach the robot the skills for every needed task and for every foreseen situation, since the number of demonstrations the human must provide

D. Hernández García (✉) · C. A. Monje · C. Balaguer
University Carlos III of Madrid, Madrid, Spain
e-mail: dhgarcia@ing.uc3m.es

to the robot to generate a new model of a skill could turn it into a tiresome and time-consuming process; furthermore, it would not be possible to cover every required task and every situation. Therefore, it is necessary to allow the adaptation of previously learned motion skills to new unseen contexts. The models of a skill are adapted to generate a new task by a merger, transition, combination or update operation over the given robot skills.

To reproduce a task adapted for an unseen context the robot must be given knowledge of the state of the environment and the constraints of the task. Using both the already learned model of a skill and the extracted constraints information of the current task, the model of the skill can be adapted to reproduce the task. The robotic systems must be able to store and later retrieve and use their knowledge of learned skills. The aim would be to have a knowledge base of the robot available skills for reproduction. The knowledge base needs to hold all necessary information for reproduction of the skills. The knowledge of the task would be distributed among the representation of objects, actions and events of the task and the state of the world.

This work is centred around the major idea of future robotic systems, more specifically humanoid robots, with the cognitive capabilities that allow them to interact with humans in their homes, workplaces, and communities, providing support in several areas, and to collaborate with humans in the same unstructured working environments. Our focus is on topics concerning the learning, representation, generation and adaptation, and reproduction of robot skills knowledge. In this work an architecture is proposed for the learning, generation and adaptation of robot skill models for complying with task constraints. The main contribution of this work is the implementation of all the modules composing the cognitive architecture proposed and their integration in order to provide the robot with a primary level of cognition.

The rest of the paper is organized as follows. A review of related cognitive systems is presented in Section 2. Our proposed architecture is introduced in Section 3. Section 4 addresses the learning of the skill models. Section 5 discusses the representation and organization of knowledge. Section 6 discusses the adaptation of the robot skills. Section 7 deals with the reproduction of the robot skills. The experimental validation is described in Section 8. Finally, Section 9 presents the main conclusions and future works from this work.

2 Cognitive Systems for Intelligent Robots

The humanoid robots that are expected for the future, capable of working autonomously and serving humans, are required to have advanced motor control skills, comprehensive perceptual systems, and suitable intelligence, where an intelligent agent is understood as in [33], as one that is flexible to changing environments and changing goals, and one that learns from experience and makes appropriate choices, given perceptual limitations and finite computation.

The study of the mind, intelligence, and the working processes of intelligent thought are the competencies of cognitive science. Research in cognitive architectures constitute a solid basis for building intelligent systems centered on the configuration and interaction of cognitive modules dealing with the various mechanisms and abilities that constitute the various processes of human intelligence.

The cognitive architecture function is to provide a comprehensive initial framework for the modeling and understanding of cognitive phenomena, in a variety of task domains, [38]. The architecture design must specify overall structures, essential divisions of modules and their interrelationships, basic representations, essential algorithms and a variety of other aspects. The various attempts at developing cognitive architectures can differ in the assumptions they make, and the design decisions they take about how to manage these aspects. A cognitive architecture can support several capabilities, and can differ variedly in their set of abilities.

Vernon [41], discerns among two major classes of cognitive systems along their different stances on the nature of cognition, what a cognitive system does, and how a cognitive system should be analyzed and synthesized. So we can group approaches as whether they are cognitivist approaches, emergent systems approaches and also efforts to combine the two in hybrid systems. For cognitivist systems cognition is representational; it involves computations of explicit symbolic representations about the world, abstracted by perception, to facilitate appropriate, adaptive, anticipatory, and effective interaction to plan and act in the world [41]. For most cognitivist approaches concerned with the creation of artificial cognitive systems, the symbolic representations are the descriptive product of a human designer. For emergent approaches cognition is the process whereby an autonomous system becomes viable and effective in its environment; it involves a process of self-organization through which the system continually reconstitutes itself [41]. The emergent approaches assert that the primary model for cognitive learning is anticipative skill construction rather than knowledge acquisition, in emergent approaches embodiment and the physical instantiation plays a pivotal role in cognition. The critical distinction between cognitivist and emergent approaches is not between representational and non-representational solutions but among action-neutral form of internal representation, requiring disembodied symbolic computational processing, and action-oriented forms, in which a behavioural response is embedded into the representation itself [12]. Considerable effort has also gone into developing approaches which combine aspects of both systems. For hybrid approaches perception-action behaviors, rather than the perceptual abstraction of representations, become the focus. The ability to interpret objects and the external world is dependent on its ability to flexibly interact with it. Hybrid systems are in many ways consistent with emergent systems, while still exploiting programmer-centered representations [41].

In the field of Artificial Intelligence and Cognitive Systems there are various works on the development of cognitive architectures to model cognitive processes and functionalities of humans. We will summarize some of the better known architectures.

The Soar (State Operator And Result) [27], cognitive architecture has been under continuous development since the early 1980s. The architecture is based on the theoretical framework of knowledge-based systems seen as an approximation to physical symbol systems [13]. Soar stores its knowledge in the form of production rules, which are in turn organized in terms of operators that act in the problem space. The basic deliberative acts of the system are performed by the operators, with knowledge used to dynamically determine their selection and application [28].

ACT-R (Adaptive Control of Thought-Rational) [4], architecture is primarily concerned with modeling human behaviour. The aim is to build systems that perform the whole space of humans cognitive tasks and describe mechanisms' underlying perception, thinking and action [13]. The ACT-R architecture is organized into a set of modules, including sensory modules for visual processing, motor modules for action, an intentional module for goals, and a declarative module for long-term declarative knowledge. The ACT-R architecture has been applied in intelligent tutoring systems, psychological studies, including aspects of memory, attention, reasoning, problem solving, etc., and to control mobile robots that interact with humans [28].

EPIC, (Executive Process Interactive Control) [24], aims at capturing human perceptual, cognitive and motor activities through several interconnected processors working in parallel, and to build models of human-computer interaction for practical purposes [13]. The architecture encodes long-term knowledge as production rules, and a set of perceptual (visual, auditory, tactile) and motor processors. Research on EPIC has included a strong emphasis on achieving quantitative fits to human behavior, especially in tasks that involve interacting with complex devices [28].

RCS (Real-time Control System) [2], is a cognitive architecture, originally designed for the sensory-interactive goal-directed control of laboratory manipulators. It has evolved over three decades into real-time control architecture for intelligent machine tools, factory automation systems, and intelligent autonomous vehicles [3]. The RCS architecture consists of a multi-layered hierarchy of computational modules, operating in parallel, containing elements of sensory processing (SP), examining the current state, world modelling (WM), predicting future states, value judgment (VJ), selecting among alternatives, behaviour generation (BG), carrying out tasks, and a knowledge database (KD).

Global Workspace Cognitive Architecture [36], is a brain-inspired cognitive architecture that incorporates approximations to the concepts of consciousness, imagination, and emotion. Cognitive functions are realized through internal simulation of interaction with the environment and action selection is mediated by affect. The architecture is based on an external sensorimotor loop and an internal sensorimotor loop in which information passes though multiple competing cortical areas and a global workspace [41].

Cog: theory of mind [34], focuses on social interaction as a key aspect of cognitive function. Cog is an upper-torso humanoid robot platform for research on developmental robotics. Cog has a pair of six degree-of-freedom arms, a three degree-of-freedom torso, and a seven degree-of-freedom head and neck. The Theory of Mind focus is on the creation of the precursor perceptual and motor skills upon which more complex theory of mind capabilities can be built. A robot possessing a theory of mind would be capable of learning from an observer using normal social signals and would be capable of expressing its internal state though social interactions, recognizing the goals and desires of others and anticipate the reactions of the observer and modify its own behavior accordingly [41].

Other attempts to provide cognitive processes and functionalities for a humanoid robot can be found in [7, 8, 11, 14, 20, 25, 26, 29, 43], and [39], among others. Efforts in cognitive architectures have produced important advances in cognition, reasoning and conceptual aspects of human thinking. [30] offers an overview of the challenges and efforts taken in the subject of cognitive robotics. [41] offers a very complete survey of artificial cognitive systems and their implications for the development of computational agents. A review of various different cognitive architectures, issues and challenges, can be found in [28].

3 Cognitive Architecture for Task Oriented Control

Research into cognitive architectures is important to improve the control and design of the intelligent robotic agents. Here we consider as a cognitive architecture: the minimal configuration of a system that is necessary for the robotic platform to exhibit cognitive capabilities and behaviors, the specification of components of the system, their function, and their organization as a whole, as defined in [41].

The desired cognitive agents must display capacities for environmentally coupled embedded action: and at the same time, they must think or reason abstractly about the world in a de-coupled manner, as argued by the theories of embodied situated cognition. Perception and recognition, decision making, memory, and learning are the most central abilities an architecture must support to cover the range of human-level intelligence. Other relevant abilities are those of problem solving and planning, prediction, reasoning, communication and action execution [28]. At the very least a cognitive architecture must present some mechanisms, structure, and organization which allow the system to be autonomous and effectively act to a limited extent [41].

In [1] a multi-layered hierarchical system architecture is proposed, where different levels of intelligence in the hierarchy can be achieved, depending on the computational power of the system and the sophistication of its processing algorithms. A minimal level of intelligence requires at least the ability to sense the environment, make decisions and take actions. Higher levels of intelligence may include the ability to recognize objects and events, to represent knowledge in a world model, and to reason about and plan for the future. More elevated forms of intelligence provide the capacity to perceive and understand, to choose wisely, and to act successfully under a large variety of circumstances.

The current humanoid robots may only be around the minimum and mid-levels of intelligence. Even if perhaps the ultimate levels of intelligence could turn out to be out of reach, and creating robots that replicate the total scope of human intelligence may prove impossible, it is necessary for future humanoid robots to achieve a sufficiently high level in the hierarchy. A cognitive architecture for humanoid robots needs to provide a minimum degree of intelligent behaviour; this is, the ability to sense the environment, learn, and adapt its actions to perform successfully under a set of circumstances.

The reference model architecture [1, 3] identifies five elemental systems: sensory processing, world modelling, behaviour generation, value judgement and knowledge, interconnected in a way that enables the various system elements to interact and communicate with each other in intimate and sophisticated ways.

Research efforts must focus on building the necessary modules of cognition that would form the layers in this hierarchy and allow the assembling of the levels of intelligence. The work in [41] summarizes some of the key features that an artificial cognitive system should exhibit, such as a minimal set of innate behaviors, a physical instantiation, means to adapt, and mechanisms for perception, action, adaptation, anticipation, and motivation that enable its development over time.

It becomes apparent that humanoid robots must be provided with systems that allow them to continuously learn new skills and adapt their existing skills to new contexts, as well as to robustly reproduce new behaviours in a dynamical environment in order to cope with working in continuously changing environments and performing an unlimited variability of tasks.

Figure 1 illustrates our proposed architecture. The main purpose of the architecture is to provide the humanoid robot with a basic level of intelligence, namely, the ability to sense the environment, learn and adapt its actions to perform successfully under a set of circumstances. In the developed architecture a knowledge base of skills is built with the models of the skills learned through demonstrations. During execution the constraints of a requested task are extracted from the perception of the working environment and the models of an appropriate skill are retrieved from the skills knowledge base. With all available information a new adapted task model is generated for reproduction.

The proposed architecture is formed by 4 fundamental modules:

1. Module for the learning of robot skills.
2. Module for the representation and management of robot skill knowledge.
3. Module for the generation and adaptation of robot skill models.
4. Module for the reproduction of robot skills.

The robot skill learning module collects the learning processes and algorithms used for learning and encoding the models of the skills. The robot skill knowledge module controls the developed knowledge base for the storing and retrieval of the learned models of the skills. The robot skill generation and adaptation module governs the process by which the learned model of a skill can be operated to reproduce a new task. The robot skill reproduction module produces the adequate control signals to the robot for the reproduction of those skills. Additionally, a perception and interaction module is in charge of processing the outside information of the robot working environment to be used by the other modules. First interactions of this architecture have been presented in [16], but further steps towards the real integration and validation of the whole architecture are presented here, this being the main contribution of this work.

The ultimate goal for a humanoid robot would require it to present full level cognitive and intelligent architectures, yet current developments are not even close to these capacities. The cognitive architecture archetype could, eventually, very well be the most suitable approach for building the humanoid robots' intelligence capabilities. However, a majority of current cognitive approaches focus more on solving intelligence as an abstract reasoning process and do not take into account the physically embedded aspects of cognition and the particular challenges humanoid robotics represents. Furthermore, fully developed cognitive architectures with the capabilities for endowing robots with the needed functional intelligence are not readily available. Therefore we begin our approach by trying to attain a basic functional level of intelligence allowing a robot to sense the environment, learn, and

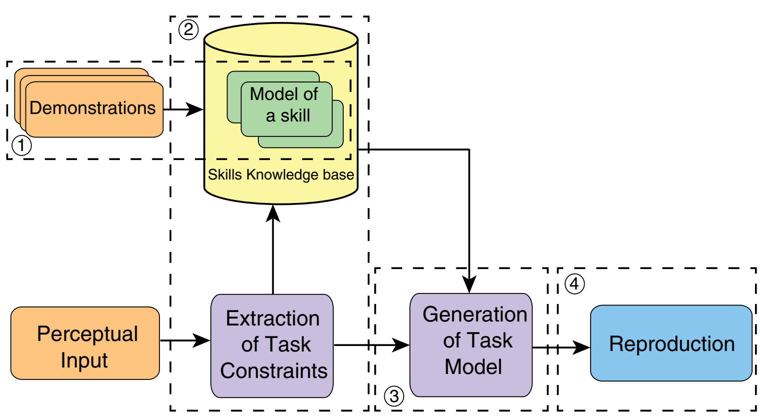


Fig. 1 Proposed cognitive architecture for task oriented control. A knowledge base (2) is built with the models of the robot skills learned through demonstrations (1). The constraints of a requested task are extracted from the perception of the world state. With the current task constraints and the models of a skill retrieved from the knowledge base an adapted task model (3) is generated for reproduction (4)

adapt its actions to perform successfully under a set of circumstances.

3.1 Implementation of Task Oriented Cognitive Architecture

A deeper discussion on the implementation of the architecture proposed is given in this section. Figure 2 shows the architecture modules and their interconnection in detail. The architecture is formed by modules for the learning of robot skills, the perception and interaction with the environment, the management and representation of skill knowledge, the generation and adaptation of skill models, and the reproduction of robot skills.

The robot skill learning module collects motion data from demonstrations, processes it and builds the demonstration data set that feeds the learning algorithms. There are three subsystems in this module: a subsystem for gathering demonstration data; a subsystem for building an estimate of the demonstration with the learning algorithm; and a subsystem for encoding the robot skill model. The subsystem for gathering demonstration data is made up of three processes. First a teacher agent input data is collected. Second, a preprocessing step is performed to transform the collected data to ensure correspondence with the robot system. A final third step processes the raw data from the previous step to build the demonstration data set as required to feed the learning algorithm. The operation of the subsystem for gathering demonstration data is handled by an external processor with different implementations for the recording of the teacher demonstrations. The learning algorithm subsystem handles the learning of the robot skill. The subsystem for encoding the robot skill model is in charge of preparing and expressing the learned estimates of the motions as *Robot Skill Models* for the rest of the architecture. The learning process is carried out off-line. The implemented system is derived from the *SEDS* library provided by [23], as describe in Section 4.

The robot skill knowledge module governs the operation of the knowledge base and the instantiation and maintenance of the different frames in the developed knowledge representational structure. There are three subsystems in this module: a subsystem for the data entry to the knowledge base; a subsystem for the knowledge base data storage; and a subsystem for the knowledge base data management. The knowledge base data entry subsystem works as a middleware between the knowledge base data storage subsystem and the robot skill learning module for uploading robot skills models and action, object and task classes for storage into the knowledge base. The knowledge base data storage subsystem works as a database collecting and organizing the robot skill knowledge as per the representational structure discussed in Section 5. Entries in the knowledge base are implemented using the *XML* markup language, following the structure and tag labels as necessary for the different knowledge frames. The physical implementation of the knowledge base is on an accompanying PC outside of the robot main system. Communications with the robot on-board computer are carried out using a WLAN

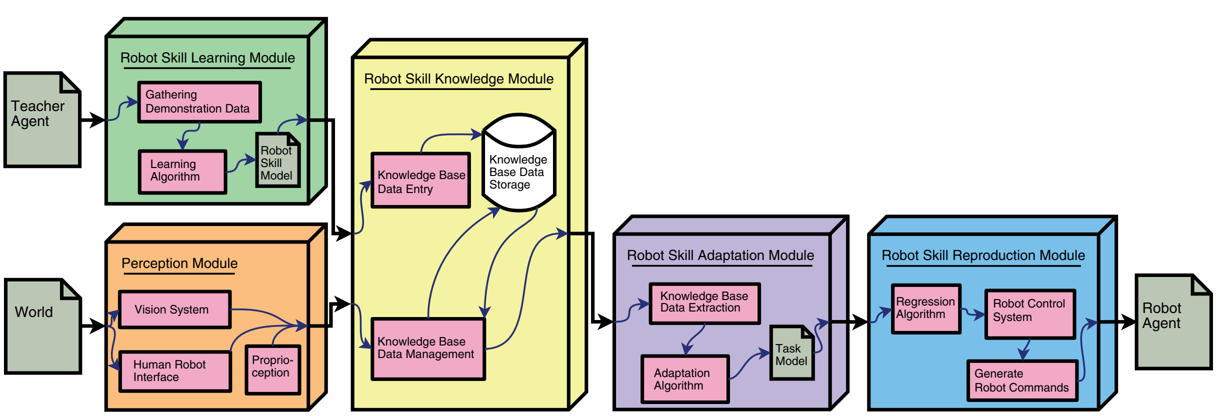


Fig. 2 Deployment diagram for the proposed cognitive architecture. The architecture is formed by a robot skill learning module, a perception and interaction module, a robot skill knowledge module, a robot skill generation and adaptation module, and a robot skill reproduction module

network. The knowledge base data management subsystem handles the operation and performance of the knowledge base. In the knowledge base data management subsystem, search and reasoning operations over the stored knowledge are carried out. A YARP layer was implemented for the communications between the robot skill knowledge module and the rest of the systems.

The robot skill generation and adaptation module is in control of handling the process by which learned models of a skill are adapted for an unseen context. This module is provided with knowledge of the state of the environment and the constraints of the task extracted from the robot skill knowledge module; using both, the already learned model of a skill and the extracted constraints information of the current task, the model of the skill is adapted to reproduce the task. There are three subsystems in this module: a subsystem for extracting data from the knowledge base; a subsystem for operating upon the robot skill with the adaptation algorithm; and a subsystem for generating the task models. The subsystem for extracting data from the knowledge base is made up of two processes: first, it recovers data from the robot skill knowledge module, and second, it distributes appropriately this data to the rest of the subsystems for their operations. This subsystem implements a middleware between the knowledge base and the rest of the systems. The adaptation algorithm subsystem handles the process of operating upon the learned robot skills. A first step from the information received from the previous subsystem would help it decide which type of method is required for adaptation; afterwards the chosen algorithm would work on the given robot skill models as described throughout Section 6. The subsystem for generating the task models is in charge of preparing and expressing the adapted *Robot Skill Models* in a form suitable for robot reproduction. As a final step, a file is outputted storing the computed task model.

Obviously all efforts in our architecture would be useless if the robot was not equipped with proper mechanisms for the motor control of the robot skill reproduction. The robot skill reproduction module is in charge of producing the adequate control signals to the robot for the reproduction of robot skills as described in Section 7. This module has three subsystems: a subsystem for computing regression of the model with *GMR* to obtain the desired target commands; a subsystem for producing the adequate control signals from the target commands; and a subsystem to communicate the control signals to the robot and monitor the *HOAP-3* robot execution.

4 Learning Robot Skills Models

The robot skills models are learned by employing an autonomous dynamical systems (*DS*) approach. *DS* has been proposed representing movements as mixtures of non-linear differential equations with well-defined attractor dynamics [19]. Common approaches in learning from demonstration create a model of the skill based on sets of demonstrations performed in slightly different conditions generalizing over the inherent variability to extract the essential components of the skill [9]. Employing statistical learning techniques is a popular trend for dealing with the high variability inherent to the demonstrations [6].

4.1 Learning Motion Dynamics as Multivariate Gaussian Mixtures

The *DS* framework provides an effective mean to encode trajectories through time-independent functions that define the temporal evolution of the motions. First it is assumed that the state of the robot system can be unambiguously described using a state variable defined as ξ (end-effector positions, velocities, etc.), and further assumed that the motion is governed by a first order autonomous ordinary differential equation

$$\dot{\xi} = f(\xi, \theta) \tag{1}$$

A probabilistic framework is employed to build an estimate $\hat{f}$ of the non-linear state transition map f, based on the set of demonstrations. *Gaussian Mixture Models (GMM)* are used to directly embed the multivariate dynamics of a motion through the encoding of the demonstrated data. A mixture model of **K** components is defined by a probability density function

$$p(\xi) = \sum_{k=1}^{\mathbf{K}} p(k)p(\xi \mid k) \tag{2}$$

where ξ is a data point, $p(k)$ is the prior and $p(\xi \mid k)$ is the conditional probability.

The *GMM* defines a joint probability distribution $p(\xi^i, \dot{\xi}^i)$ of the training set of demonstrations as a mixture of the **K** Gaussian multivariate distributions:

$$p(\xi, \dot{\xi}; \theta) = \frac{1}{K} \sum_{k=1}^{\mathbf{K}} \pi^k \mathcal{N}^k(\xi, \dot{\xi}; \mu^k, \Sigma^k)$$
$$\text{with} \quad \mu^k = \left\{\mu_\xi^k; \mu_{\dot{\xi}}^k\right\} \quad \text{and} \quad \Sigma^k = \begin{bmatrix} \Sigma_\xi^k & \Sigma_{\xi\dot{\xi}}^k \\ \Sigma_{\dot{\xi}\xi}^k & \Sigma_{\dot{\xi}}^k \end{bmatrix} \tag{3}$$

The *GMM* estimates the function f, so the unknown parameters θ are the prior, π^k, the mean, μ^k, and the covariance matrix, Σ^k, of the **K** Gaussian, such that $\theta^k = (\pi^k, \mu^k, \Sigma^k)$, as in Eq. 3, defines the robot skills models.

To recover the expected output variable $\hat{\dot{\xi}}$, given the observed input in ξ^*, one then can sample from the probability distribution function $p(\xi, \dot{\xi})$ in Eq. 3. This process is called *Gaussian Mixture Regression (GMR)*. Taking the conditional mean estimate of $p(\dot{\xi} \mid \xi^*)$ the estimate of our function $\hat{\dot{\xi}} = \hat{f}(\xi^*)$ is:

$$\hat{\dot{\xi}} = \sum_{k=1}^{K} h^k(\xi^*) \left(\Sigma_{\dot{\xi}\xi}^k \left(\Sigma_\xi^k\right)^{-1} \left(\xi^* - \mu_\xi^k\right) + \mu_{\dot{\xi}}^k \right) \tag{4}$$

$$\text{where} \quad h^k(\xi) = \frac{p\left(\xi; \mu_\xi^k, \Sigma_\xi^k\right)}{\sum_{k=1}^{K} p\left(\xi; \mu_\xi^k, \Sigma_\xi^k\right)} \text{with}$$
$$h^k(\xi) > 0 \quad \text{and} \quad \sum_{k=1}^{K} h^k(\xi) = 1$$

4.2 Stable Estimator of Dynamical Systems

The work in [22, 23] proposed a learning method, called *Stable Estimator of Dynamical Systems (SEDS)*, to learn the parameters of the *DS* ensuring asymptotically stable trajectories for all motions that closely follow the demonstrations dynamics. In this work we follow their framework to learn the motions as multivariate *DS* within a Learning from Demonstration (*LfD*) statistical approach.

The unknown parameters θ must be determined such that by starting the motion from any point the energy of the system decreases until it reaches the target. Learning the parameters of the *GMM* proceeds as a constraint optimization problem, ensuring that the model satisfies global asymptotic stability of the *DS* at the target [23]. By being time-invariant and globally asymptotically stable at the target, the *DS* estimated with *SEDS* are able to respond immediately and appropriately to perturbations that could be encountered during reproduction of the motion [23]. The process is illustrated in Fig. 3 for the teaching of a skill with the humanoid robot HOAP-3.

5 Representation of Robot Skills Knowledge

An important challenge for robots acting on unstructured dynamic environments is dealing with internal representation and understanding of the world. Decisions must be made on which aspects of the world to focus on and which aspects to ignore, and how to structure that knowledge.

The interrelation between objects and actions representations is fundamental when executing tasks upon the world. Thus focussing only on objects and actions would not be enough for the knowledge representation needed by the humanoid robots. Representational attributes need to also take into account the state of the world, grounding the representations to the environment, the task at hand and present events. The central task of a knowledge representation is capturing the complexity of the real world. Representations thus perform as functional abstractions of the perceived environment, encoding an agent's knowledge about its world, objects, actions, events, into manageable internal structures.

A system dealing with objects in the real world must deal with various forms and types of knowledge. Here we will organize our knowledge into manageable structures using object-oriented groups of procedures, which are called frames. Representations of events concentrate on two frames, one of the system tasks knowledge and the other representing the state of the world knowledge. Task and world frames would hold the knowledge of the requested execution of a task and the agent's environment.

5.1 Knowledge Base Structure

The knowledge base needs to hold all necessary information for reproduction of the skills in the environment. The knowledge of the task would be distributed

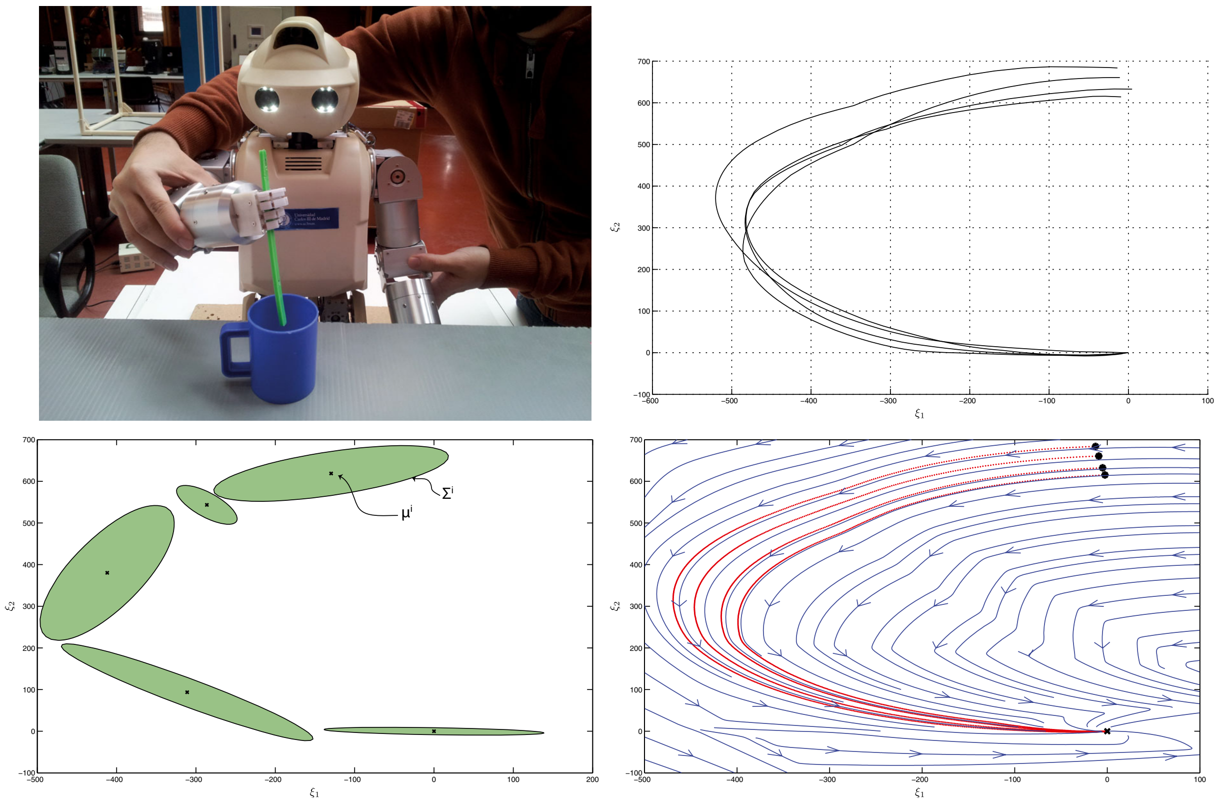


Fig. 3 Illustration of the learning process with *GMM-GMR*. (*top-left*) A human teacher operates the HOAP-3 robot arms through a demonstration of the skill. (*top-right*) Recorded training data of the demonstrated trajectories. (*bottom-left*) The learned *GMM* model represented by ellipses centred at μ^i, magnitude and direction of the ellipses are given by the eigenvectors and eigenvalues of Σ^i. (*bottom-right*) Reproduction of several trajectories through *GMR*

among the representation of objects, actions and events of the goal and the state of the world. From a given scene the system instantiates frames, generally governed by the precedence of visual evidence. From the perceived given input the first step for extracting task constraints is the matching of the world to an instance of the World Event Frame and the instantiation of the Task Event. From the information collected in the World and Task event frames, which in turn are made up by Object and Action Frames, the system would have information about its current goals and situation of the environment, yet this is not enough to ground the representation in order to effectively use it for supporting the robot performance. For an agent working in an unstructured environment the focus of its perception must be directed towards its executing action. Knowledge of its environment and task would be collected into their appropriate frames and a focused active view frame would be built taken from their global knowledge and breaking it down into a simpler framework from which computations and knowledge take place. Figure 4 presents the control data flow for the process of using the representations in the knowledge base and Fig. 5 presents the organization of the knowledge base.

In order for a process to use a representation, the process must be coordinated with the format of the representation, only states appropriate to the process will count as representations [5]. The process for using the representations begins by instantiating the appropriate frames. The data structure of a frame is made up of slots filled with attributes, which can be made of other frames, as organized in terms of a class hierarchy, analogous to an object-oriented programming paradigm. When instantiating a frame its slots will be filled with the values present in the system, any slots with unavailable information will be filled by default attributes associated with the class categories.

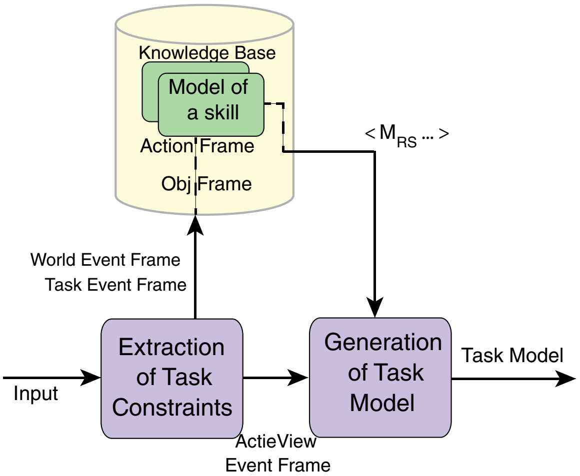


Fig. 4 Knowledge base control flow. World Event Frame and Task Event Frame are instantiated, and an Active View Event Frame is built from them. From Object and Action Frames the models of the skill are taken for building the task model

Default values are assumptions reasonably made when the state of knowledge holds no information to the contrary.

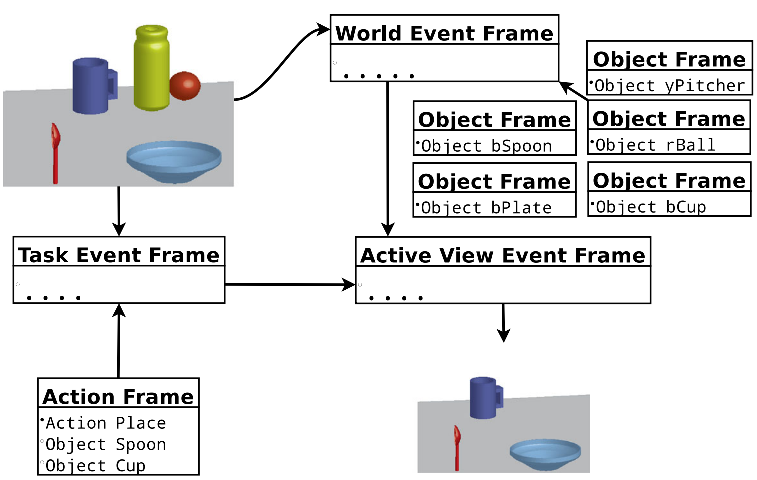
Figure 6 shows the representation of the skills in the knowledge database in a three dimensional space defined by the ⟨*Object, Goal, World State*⟩ triple, selecting from their intersection an adequate model of the skill for the reproduction of the task.

Further development of the knowledge base representation and structure can be found in [18]. Different approaches on related topics focusing on the management of knowledge by robotic system exist, such as KnowRob [40] or RoboEarth [42]. However these systems lie at a higher more abstract level of the cognitive hierarchy while our framework lies at a lower level of action execution. Further research requires study and comparison of other systems, in particular the ones that may be used to complement the framework developed in this work.

6 Adaptation of Robot Skills

The robot skills learned with the methodology described in Section 4 would present stable trajectories that accurately reproduce the demonstrated motion dynamics. These learned models would form a set of primitives of action from which a knowledge base of skills was built in Section 5. Evidences exist from human and animal experiments supporting the believe that sets of motor primitives are used to build a basis for voluntary motor control [35]. To generate complex motions from a learned set of basic primitive skills and be able to reproduce various complex task behaviours, methods for operating and manipulating upon the primitives are needed. The robot skills must be adaptable to conditions of its operating environment. The models of a robot skill must be updatable: when given new information for the representation of a skill the system must allow the models to be improved. Additionally, the action primitives approach must be able to generate new skills by merging two or more primitives into a new skill: multiple desired robot skills may be composed from

Fig. 5 Knowledge base structure and organization of the knowledge representations. World Event Frame and Task Event Frames represent the knowledge of the state of the environment, with Object and Action Frames representing the available objects and actions. From the knowledge of these frames an Active View Event Frame is built of the focused knowledge required to drive the agent execution



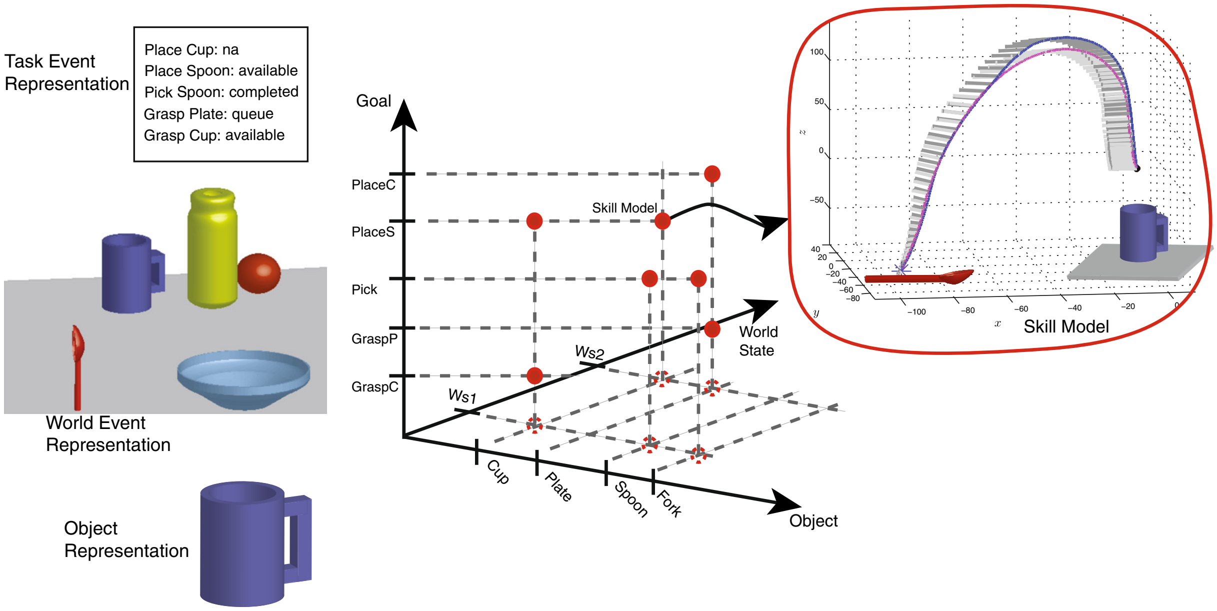


Fig. 6 Representation of the skills in the knowledge base. The intersection of the triple ⟨*Object, Goal, World State*⟩ allows the selection of the adequate model of the skill for reproduction

superposition of various primitives. Another important property is the combination of the robot skills models to generate new models that encompass a larger spectrum of the attractor dynamics. Our previous works in [15, 17] contain more detailed information about generation and adaptation of robot skills models, and a short review is given next.

6.1 Update of Robot Skills

When an update is required with new given data, a process of *GMR* regression is performed over the learned model to stochastically generate a dataset from the model. Therefore a new dataset is created composed of this generated demonstration and the new observed dataset. The parameters of the updated model are then retrained. For this purpose a learning rate α is defined.

For our method the new updated demonstration dataset $\{\xi, \dot{\xi}\}_{updated}$ is grouped into $\mathbf{K}$ clusters according to the number of Gaussian functions determined for the original robot skill model. Parameter α is defined as $\alpha^k \in [0; 1]; k = 1..\mathbf{K}$, and it determines a measure of the relative importance of the area in cluster k the updated demonstration should have for refining the model over the stochastic demonstrations generated from the learned model.

To illustrate this method Fig. 7 shows the result of updating a learned model of a skill. The updating process is summarized in Table 1.

6.2 Merger of Robot Skills Models

Intuitively one could consider an approach the fact of merging two or more models of a skill simply by adding and averaging together their learned parameters $\theta = (\pi, \mu, \Sigma)$ in order to obtain a new skill model. While this approach may work for some cases, it is important to note that the direct superposition of the skills does not allow the system to control the manner in which the new model is generated and its stability.

In order to generate a new skill based on the merger of several robot skills previously learned, we first review a couple of useful mathematical properties from the *SEDS* [23] formulation chosen to learn the skills:

$$\begin{aligned} &\text{if} \quad f(\xi) \quad \text{is } SEDS\text{, and} \quad \alpha > 0 \in R \\ &\dot{\xi} = \alpha f(\xi) \quad \text{is } SEDS \\ &\text{consider} \quad \mathbf{M} \quad SEDS \text{ models} \quad f^i(\xi), i \in 1..\mathbf{M} \\ &\dot{\xi} = \sum_{i=1}^{\mathbf{M}} \alpha^i f^i(\xi); \alpha^i > 0 \quad \text{is } SEDS \end{aligned} \tag{5}$$

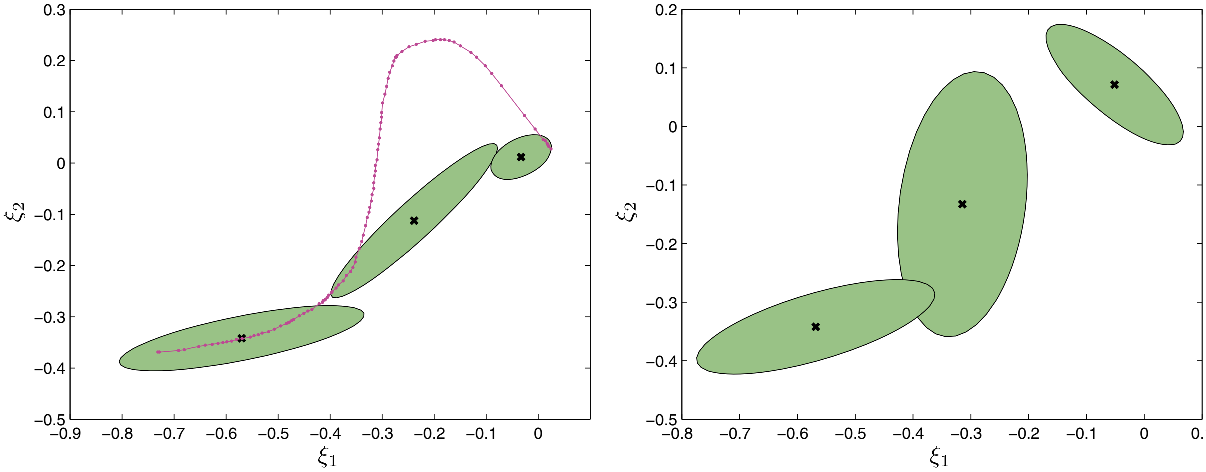


Fig. 7 Update process of a robot skill. (*left*) Model of the learned skill with new demonstrations. (*right*) Updated model of the skill. The parameter α^k is defined to govern the influence of new data on the update process. Appropriate selection of α^k allows the updated model to reproduce the curve at the top of the trajectory

The merger of the robot skills can be carried out with a model combination approach expressed as mixtures of expert models:

$$p(t|x) = \sum_{k=1}^{\mathbf{K}} \pi_k(x) p_k(t|x) \tag{6}$$

The *SEDS* models encoded into a *GMM* are already a form of model combination approach. Here, recalling the expression of the non-linear weighting function $h^k(\xi)$, as in Eq. 4, it can be found that it shares a similar formulation with the expression of the weights for the gating function as from Eq. 6. The process for the merging of robot skills would first join the *GMM* of the robot skills into a single model. Then a new weighting function $\tilde{h}(\xi)$ for the single model must be built out of the original weighting terms $h^k(\xi)$ from the merged models, ensuring that the Gaussian with the biggest weight in every region of the trajectory provides the largest influence over the new *GMM* model in that region and that the new weighting function $\tilde{h}(\xi)$ still meets the constraints $0 < h^k(\xi) < 1$ and $\sum h^k(\xi) = 1$.

Figure 8 illustrates the results of merging two robot skills to generate a new skill model. The merging process is summarized in Table 2.

6.3 Combination of Robot Skills Models

In order to generate a new skill made of the combination of several robot skills models previously learned, we have developed a method for skills combination. Two different *SEDS* models, $\bar{\mathcal{M}}^1_{RS}$, $\bar{\mathcal{M}}^2_{RS}$, can be combined just by concatenating their parameters, so

Table 1 Procedure for updating a learned model of a robot skill

Algorithm: Update the learned robot skill
Input: Learned *Robot Skill Model*, $\mathcal{M}_{RS}$, with parameters $\theta^k = (\pi^k, \mu^k, \Sigma^k)$.
1. Record new demonstration trajectory for the update of the skill.
2. Generate $n_g en$ trajectories stochastically from the current model by the *GMR*.
3. Determine parameter $\alpha = \alpha^k \in [0; 1]$; $k = 1..\mathbf{K}$.
4. Create a new updated demonstration dataset $\{\xi, \dot{\xi}\}_{updated}$.
5. Generate the new updated model of the skill.
6. **END**
Output: Updated *Robot Skill Model*, $\mathcal{M}_{RS_{updated}}$, with parameters $\theta^k_{updated} = (\pi^k, \mu^k, \Sigma^k)$.

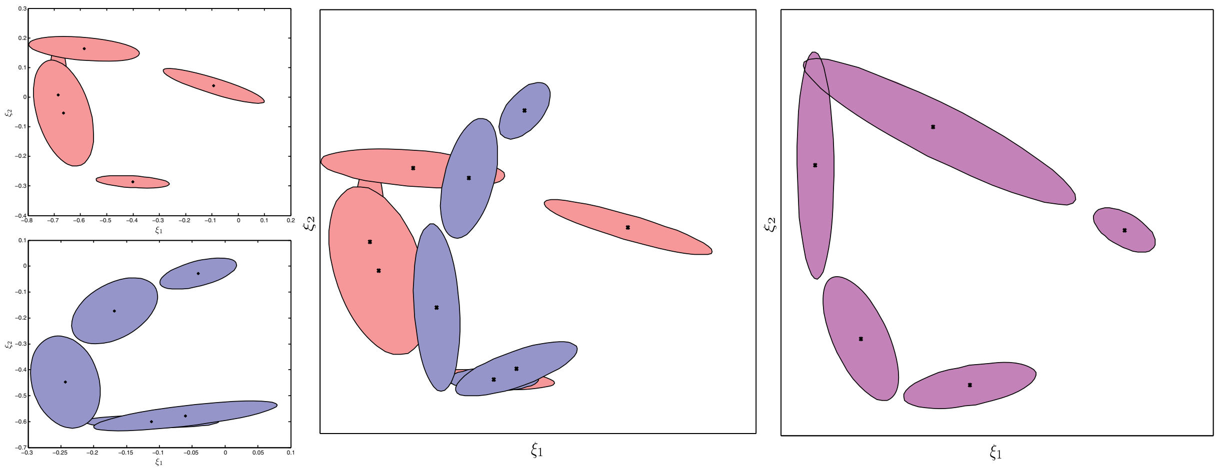


Fig. 8 Merger of two learned *GMM* skill models to generate a new skill. (*left*) Learned models of the robot skill. (*center*) Merging process of the two models to generate a new one. (*right*) Merged skill model

that the parameters of the new model can be defined as $\pi = \dfrac{[\pi^1; \pi^2]}{(\pi^1 + \pi^2)}$, $\mu = [\mu^1 \mu^2]$ and $\Sigma = [\Sigma^1 \Sigma^2]$. Then, an area of influence for the *DS* attractor is defined based on the non-linear weighting function $h^k(\xi)$ of the *SEDS* models expressed as a non-linear sum of linear dynamical systems as in Eq. 4. A new weighting function $\tilde{h}(\xi) = \alpha^k(\xi, h)h^k(\xi)$ for the single model must be built out of the original weighting terms $h^k(\xi)$, as in the merging of the models. However in this case the influence of the $h^k(\xi)$ terms over the trajectory must come at any time from only one model; therefore the $\alpha^k(\xi, h)$ function must have a completely different form from that of the merging of robot skill models.

Figure 9 illustrates the results of combining three robot skills to generate a new skill model. The combination process is summarized in Table 3.

7 Reproduction of Robot Skills

In this section, the development and operation of the robot skill reproduction module will be presented. The robot reproduction module is assigned with the task of providing suitable controllers that convert kinematic variables into appropriate motor commands. In order to test the proposed architecture the *HOAP-3* humanoid robot was used as a test platform. The *HOAP-3* was designed to resemble the human shape, on a small scale, with a complete humanoid configuration with two legs and arms, a head with vision and sound capacities, and grip-able hands.

Figure 10 presents the control strategy of the robot skill reproduction module, for details see [32]. This scheme considers several blocks. Once a command has been received, the robot distinguishes if it is a command for the walking generation or for the arms

Table 2 Procedure for merging learned skill models

Algorithm: Merger of learned robot skills
Input: Learned *Robot Skill Models*, $\mathcal{M}^1_{RS}$, $\mathcal{M}^2_{RS}$, ..., $\mathcal{M}^n_{RS}$.
1. Compute the new model as $\sum_{k=1}^{K} h^k(\xi)(\mathbf{A}^k\xi + \mathbf{b}^k)$.
2. Compute the parameters α^k for the new model.
3. Build the weighting function $\tilde{h}$, as $\tilde{h}(\xi) = \alpha^k(\xi, h)h^k(\xi)$.
4. Generate the new merged model of the skill.
5. **END**
Output: Merged *Robot Skill Model*, $\mathcal{M}_{RS_{merged}}$, given by $\sum_{k=1}^{K} \tilde{h}^k(\xi)(\mathbf{A}^k\xi + \mathbf{b}^k)$.

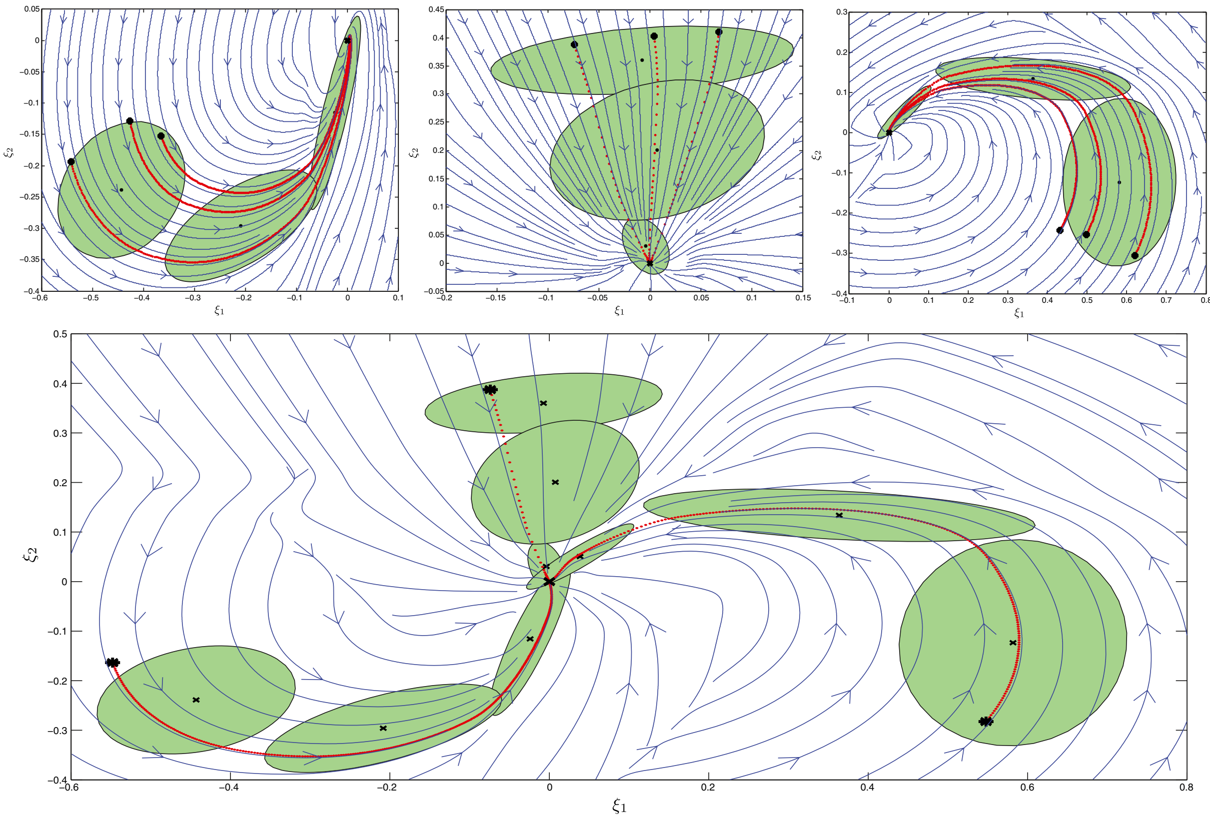


Fig. 9 Combining the dynamics of three skills into a single task model

movement. The walking patterns of the robot have been designed based on the theory of the 3D Linear Inverted Pendulum Mode presented in [21, 31] presents studies for the posture stability control. If the received command requires a movement of the arms, as in the case of a grasping task, the selection of the suitable arm is first considered. Finally, the trajectory of the arm is evaluated online through the algorithm of kinematic inversion [37], once the command provides the distance and the orientation from the object. The orientation reference for the object is calculated with the support of the unit quaternion presented in [10].

The *HOAP-3* control system is in charge of computing the appropriate command to control the execution in real-time. The physical implementation of the robot control system is made on three PCs: an on-board PC implements the robot control systems; an auxiliary PC implements the knowledge and learning systems; and

Table 3 Procedure for combining learned skills models

Algorithm: Combination of learned robot skills

Input: Learned *Robot Skill Models*, $\mathcal{M}^1_{RS}, \mathcal{M}^2_{RS}, ..., \mathcal{M}^n_{RS}$.

1. Calculate the prior $\tilde{\pi}$, as $\tilde{\pi} = \dfrac{[\pi^1; \pi^2; ...; \pi^n]}{(\pi^1 + \pi^2)}$.
2. Calculate the mean $\tilde{\mu}$, as $\tilde{\mu} = [\mu^1 \mu^2 ... \mu^n]$.
3. Calculate the covariance $\tilde{\Sigma}$, as $\tilde{\Sigma} = [\Sigma^1 \Sigma^2 ... \Sigma^n]$.
4. Build the weighting function $\tilde{h}$, as $\tilde{h}(\xi) = \alpha^k(\xi, h)h^k(\xi)$.
5. **END**

Output: Combined *Robot Skill Model*, $\mathcal{M}_{RS_{combined}}$, given by $\sum_{k=1}^{K} \tilde{h}^k(\xi)(\mathbf{A}^k\xi + \mathbf{b}^k)$.

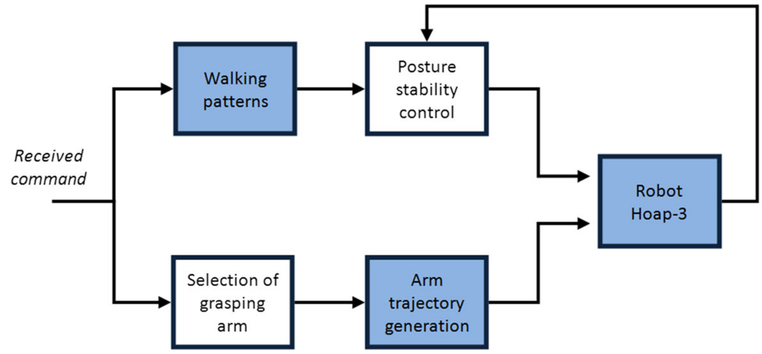


Fig. 10 Control strategy of the robot skill reproduction module

a laptop computer implements the *HRI* and perception systems. A YARP layer was implemented for the communications between processes.

8 Experimental Evaluation

Here we provide a general description of a demonstrator for the evaluation of the architecture performance. The experimental evaluations presented in this section are aimed at providing proof of concept for the developed architecture. The quantitative evaluation of the knowledge processing systems is hardly possible since many of its features are difficult to reflect in numbers. However the system can be evaluated in a qualitative form. Here the major focus of interest lies not in the measurement of performance and efficiency metrics but in the validation of the viability of the proposed system and the capabilities of the architecture in dealing with a range of different and increasingly complex situations. The demonstration will test the operation of the humanoid robot and the developed architecture as it is required to complete distinct tasks. Several experiments were conducted in order to prove the validity of the system and to highlight how the components of our architecture contribute to achieving realistic tasks, and that the implementation of the capabilities for learning, knowledge manipulation and adaptation of skills are fundamental for the development of viable humanoid robots.

8.1 Knowledge Base Scenario

A first experiment involves an agent and a humanoid robot (here a *HOAP-3* robot) interacting to complete a simple task. The task in this case requires the robot to pick up a cup and a spoon in each hand and then to put the spoon inside the cup; then finally it will put down the cup in front of it. The agent will provide the robot with the cup and spoon objects so it can pick them up; also the agent will indicate the robot where to put the cup down.

The execution of the demonstration could vary depending on the actions of both the human agent and the *HOAP-3* robot. At the start of the demonstration the robot is given the task event frame knowledge for the desired behaviour containing the knowledge of the four action skills needed to complete the action: pick spoon, pick cup, place spoon in cup, place cup down. Extracting the adequate action will depend on the agent interaction and the content of the rest of the knowledge base. The purpose of this demonstration is to validate the performance of the developed knowledge base in a dynamic interaction with an agent.

Figure 11 shows a schematic view of the overall experiment described above. The perception system handles the interaction with the user and the detection of objects in the environment. The knowledge base system would receive this information from the perception system and would instantiate the frames and build the knowledge representation of the scene in the knowledge base. The knowledge base system would select and activate an action skill when the conditions in the knowledge representation afford such action. Once an action is selected, the *HOAP-3* robot controller would execute the robot commands required for the skill reproduction. This demonstration highlights the operation of the knowledge base and how the representations of object, action, task event, world event and active view event frames are used to command the robot execution of the desired task. The goal of this demonstrator scenario is to show how action execution is invoked by the state of the representation frames

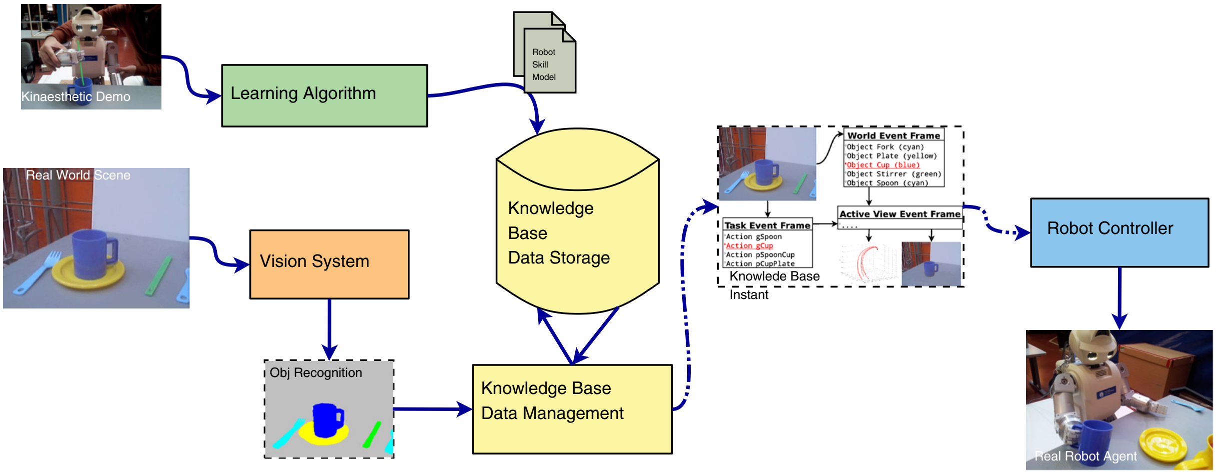


Fig. 11 Schematic view of the Knowledge Base Scenario experiment

present in the knowledge base. Figure 12 presents a storyboard of the performance of the system during the execution of the demonstrator experiments with snapshots taken at various stages.

Figure 13 presents the operation of the perception system during the execution of the demonstrator experiments. Objects are recognized based on their colour properties and blob size. From the images it can be seen that some problems can take place when the human agent or the robot platform arm enter the camera's field of view, as occlusions and false recognitions can happen. Typically, these issues can be

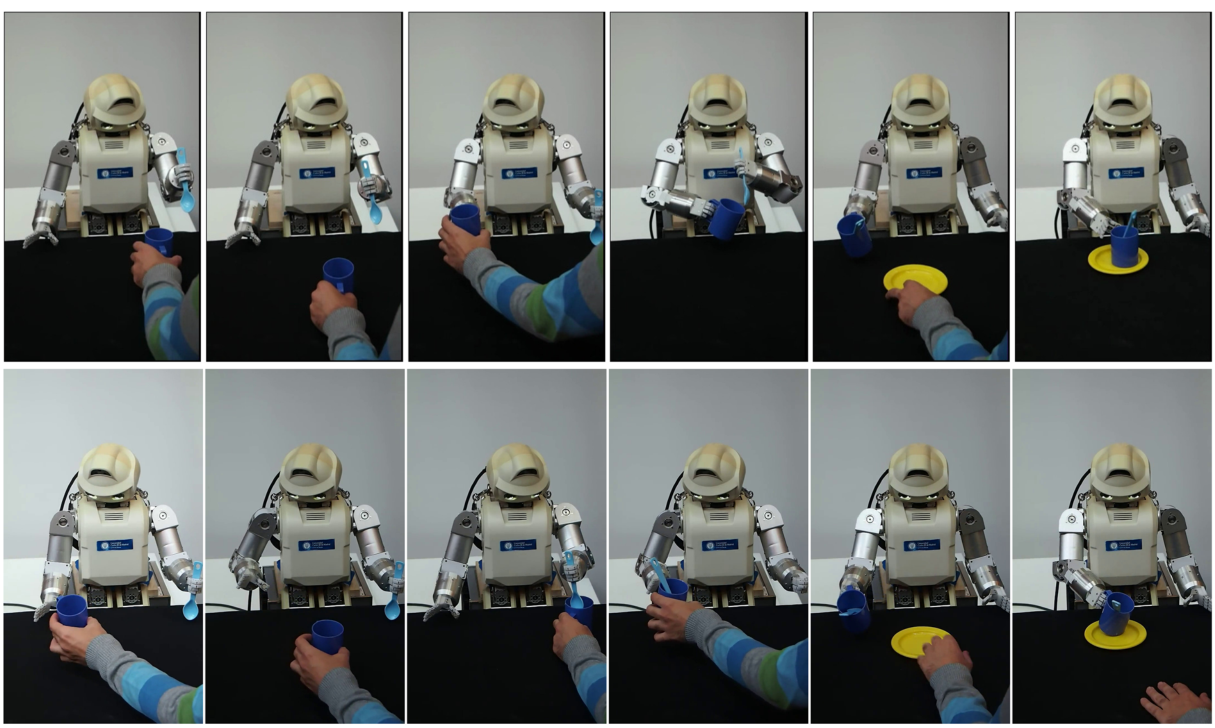


Fig. 12 Knowledge Base Scenario experiment: different snapshots during the execution of the demonstration. The top and bottom rows represent two different reproductions of the experiment

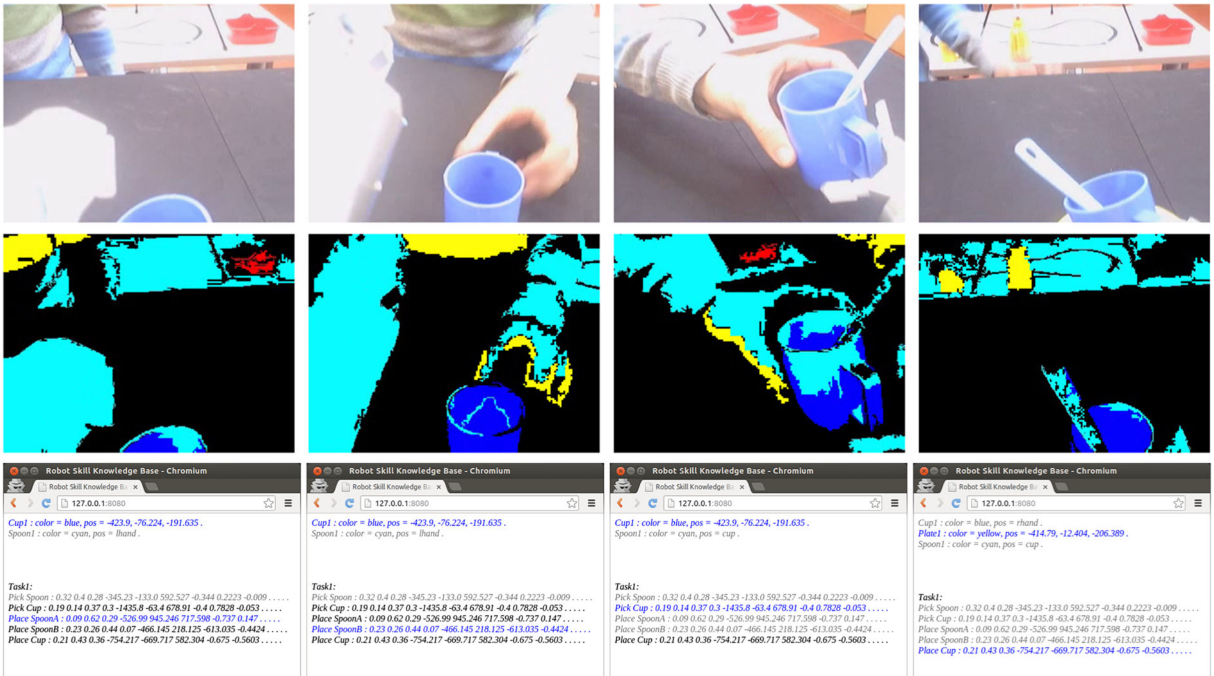


Fig. 13 Knowledge Base Scenario Experiment: different snapshots of the execution of the demonstration illustrating the operation of the perception system and the knowledge base system

taken care of by the blobs' size and area inconsistency with expected objects' properties, or by their failed instances being removed from the knowledge base.

The operation of the knowledge base system during the execution of the demonstrator experiments can be seen in Fig. 13. The knowledge base presents information for the environment and the task execution. The task frame holds knowledge of the actions to carry out by the robot for the execution of the task. Actions highlighted in blue reflect the current invocation of that action's knowledge for the robot reproduction of the skill. Actions that have been completed are deactivated and highlighted in grey. The selection and activation of which skill motion to carry out next is completely determined by the skill initial conditions being matched to the state of the environment. Therefore, the sequence of execution of the task is controlled by the human agent as it interacts with the robot and the environment and facilitates the objects and conditions needed for the robot to fulfil the task.

A potential problem is determining which action has precedence when many of them can satisfy their conditions at the same time. The tasks considered in the demonstrator do not deal with this issue, since the robot's limited workspace prevents the conditions for picking up the cup and placing the spoon to be satisfied at the same time. This issue has not been fully explored so far, and as a first simplification the precedence is determined by the order of the actions in the task frame as determined by the programmer of the task; although not satisfactory for every scenario, this solution is probable enough for many common tasks. The use of some form of long time planner could be effective to solve this issue by assigning precedence by determining how the decision of performing one action over another could affect the execution of the task.

A knowledge base approach for robots working in unstructured environments, where the execution of the task cannot be scripted beforehand, is fundamental if they are to be able to work successfully. Without such a system the robot would be unfit to respond to any unforeseen deviation from the plan, and largely ineffective to perform in all but the most ideal of situations. The knowledge base system allows the robot to keep track of the environment and the execution state of the task, which provides the system with flexibility to deal with different states at a particular point without losing focus of the global task objective.

A video of the performance of the system in this scenario has been uploaded: www.youtube.com/watch?v=3l7-KrMa84o.

8.2 Robot Skill Reproduction Scenario

As a final experiment we will visit a kitchen or dinner table scenario and expand the demonstrators presented in the previous section. In this scenario the *HOAP-3* robot is required to complete the task of setting up a dinner service together with a human agent. The purpose of the demonstrator is to test the overall operation of the developed architecture, as well as to validate the performance of every individual module and interaction between them, involving the perception of objects and interaction with the agent, the learning of various robot skills, the representation of knowledge in the knowledge base, the generation and adaptation of the skill models and the adequate reproduction of the robot skills.

Figure 14 shows a schematic view of the overall skill reproduction scenario experiment described above. For this scenario, various demonstrations of skills, recorded with the *HOAP-3* robot, are first given to the learning module to encode the models of the robot skills for the different actions required for the "dinner service" task. Subsequently, the learned robot skills are stored by the knowledge base system. During the operation, the user will provide objects to the robot by placing them in its action field, both of vision and manipulation. The perception system will handle the interaction with the user and the detection of objects in the environment. The knowledge base system will receive this information from the perception system and will instantiate the frames and build the knowledge representation of the scene in the knowledge base. Through this interaction with the user and the environment, the knowledge base system will select the corresponding skills to activate. Once the necessary robot skills are selected, the generation and adaptation system will be in charge of building the appropriate task model satisfying the desired command and constraints of the environment for reproducing the appropriate skill action. Finally, the *HOAP-3* robot controller will execute the robot commands required for skill reproduction.

This demonstrator scenario is meant to provide proof of concept of how the knowledge base operates to instantiate frames from the perception of the environment, and how the knowledge base maintains and upkeeps its knowledge representation over time in a changing environment, as well as how action execution is invoked by the state of the representation frames present in the knowledge base. Additionally, the demonstrator scenario provides validation for the generation and adaptation system and how it operates over learned robot skills for increasing the scope of available skills for the performance of the *HOAP* humanoid robot.

Figure 15 depicts a storyboard of the performance of the second demonstrator showing several snapshots captured from the execution experiment. The demonstrator scenario will develop as follows: first the robot is given the task of setting up the "dinner service" at the table in front of it, and all necessary robot skill actions and task event frames are stored in the knowledge base. The task begins with the robot standing in front of the empty table. The final set-up of the table

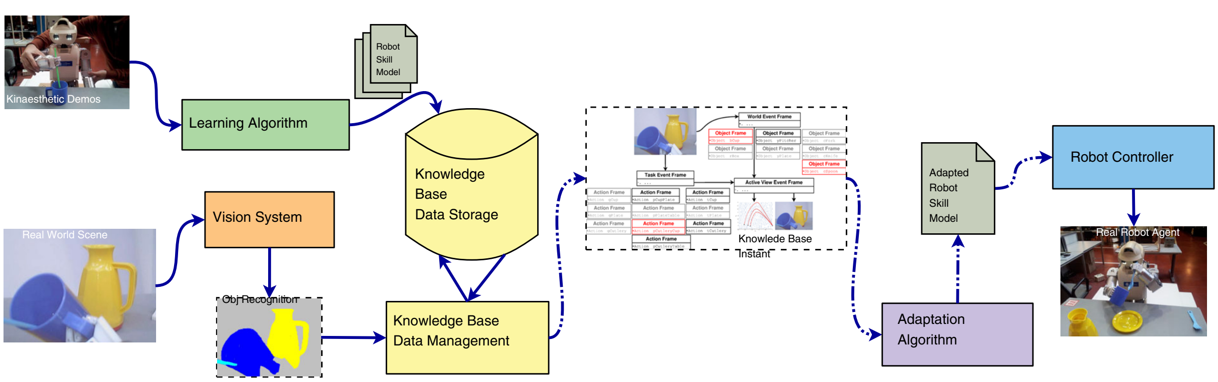


Fig. 14 Schematic view for the Robot Skill Reproduction Scenario experiment

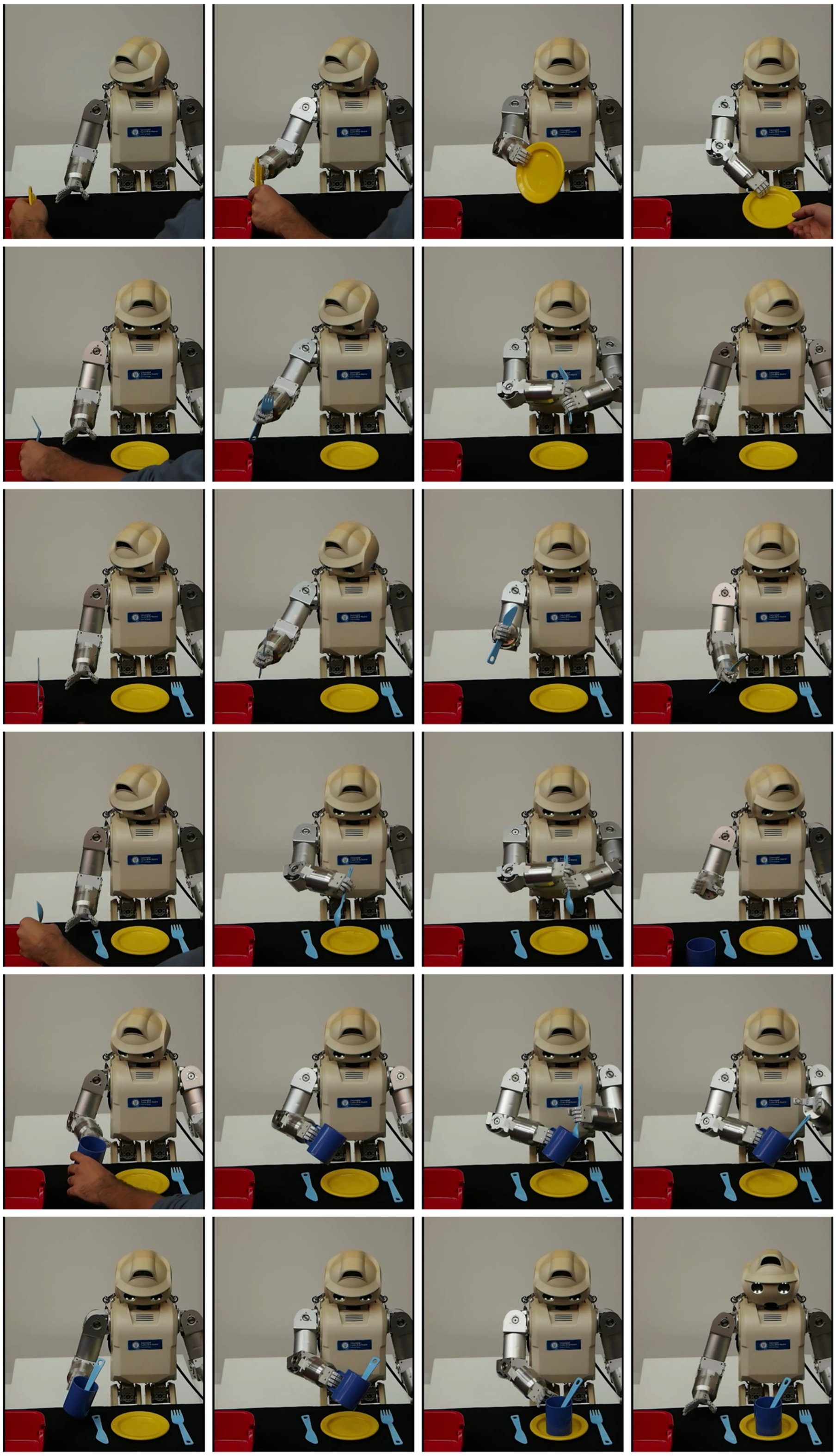


Fig. 15 Robot Skill Reproduction Scenario experiment: different snapshots from the execution of the task in the demonstrator

requires a plate to be placed in the center, a cup placed on top of the plate, a spoon placed inside the cup, and a fork and knife flanking the plate at its left and right sides, respectively. Completing the task requires the performance of several different skills. The sequence of execution of the task is governed by the human agent as it is him who chooses the order in which to provide the robot with the needed objects. Certain items, however, have precedence over others, i.e. the plate must be placed on the table before the cup, since the cup goes on top of it.

The first object to be placed on the table is a ‘red’ container box, from which the *HOAP-3* robot picks up the objects, when available. A human agent would choose from the pool of objects of the task one object to be placed by the robot. Figure 15 top left image shows the instance where the human agent sets the first object for that run of the task, in that case a plate. The *HOAP-3* robot perception would recognize the object in the container box, when this happens an object frame instance is created in the knowledge base, and the action frames in the task event frame are checked out to find which, if any, match is invoking conditions from the current state of the world frame, in order to begin reproduction of a skill. Once an action is chosen, the *Robot Skill Model* parameters $\theta = \{\pi, \mu, \Sigma\}$ are recovered from the knowledge base system and provided to the robot skill reproduction model for performing the actual reproduction of the skill, as in the *GMR* process described in Section 4. The robot will pick up the given object and carry out the required operations with it to place it correctly on the table.

A video of the performance of the system in this scenario has been uploaded: www.youtube.com/watch?v=BKXaZGV8xvM.

9 Conclusions and Future Works

This work is centred on the aspiration of building humanoid robots capable of interacting with humans in their homes, workplaces, and communities, providing support in several areas, and collaborating with humans in the same unstructured working environments. The aspiration is to have humanoid robots acting as robot companions and co-workers sharing the same space, tools, and activities.

There is much work to be done to improve the capabilities of humanoid robots for locomotion, perception, interaction, cognitive behaviour and competence at performing tasks. Humanoid robots must present intelligent, natural, predictable and reasonable behaviours, and the development of intelligent controls to resemble this is a major challenge.

The main contribution of this work is the proposition and implementation of a cognitive architecture for the generation and adaptation of learned models for task oriented control. In the developed architecture a knowledge base of the skills is built with the models of the skills learned through demonstrations. During the execution, the constraints of a requested task are extracted by the perceptual system from the working environment and the appropriate skill models are retrieved from the skills knowledge base. With all the available information, a new adapted task model is generated for reproduction.

The architecture developed in this work was proposed as a cognitive model intended to provide the robot with an essential cognitive ability for learning and adaptation of skills. Though it is not a primary consideration of this work, our architecture can be thought of as one module level in the hierarchy of a more complex architecture, or as a first stepping stone upon which to incrementally build more complex cognitive processes. The goal of the developed architecture is to provide a minimum degree of intelligence for the humanoid robot. The ultimate goal of the field calls for fully functional humanoid robots capable of performing any type of task as a human agent would, and capable of working, collaborating and interacting with humans, sharing the same space, tools, and activities. This vision requires for robots to present full level cognitive and intelligent architectures; however, current developments are not yet even close to these capacities, and our discussion needs to start at some point in a basic functional level of intelligence. We consider as a minimal desirable level of intelligence for our humanoid robots the ability to sense the environment, learn, and adapt their actions to perform successfully under a set of circumstances.

The developed architecture provides humanoid robots with systems that allow them to continuously learn new skills, represent their skill knowledge, and adapt their existing skills to new contexts, as well as to robustly reproduce new behaviours in a dynamical environment. The architecture is formed by modules for the learning of robot skills, the perception and interaction with the environment, the representation and management of the skill knowledge, the generation and adaptation of skill models, and the reproduction of robot skills. To learn the skills motion a time independent model of the motion dynamics was estimated through a set of first order non-linear multivariate dynamical systems. A knowledge base of skills has been developed and implemented. The knowledge base holds all the necessary information for reproduction of the skills in the environment. The knowledge of the task is distributed among the representation of objects, actions and events of the task and the state of the world. A structure built on frames has been adopted in this work. The knowledge of the environment and goals is represented in terms of the World Event Frame and Task Event Frames, with Object and Action Frames representing the knowledge about available objects and actions, respectively. From the knowledge of these frames an Active View Event Frame is built of the focused knowledge required to drive the agent execution.

Also, methods for models combination have been presented. Probabilistic frameworks for combining models can be viewed as mixture distributions conditioned on the input variables. The idea behind is that different components can model different regions and a gating function determines which components are dominant in which region. The manipulation of the skills must allow the adaptation, update, merger, and combination of robot skills as necessary.

The proposed architecture was demonstrated with a commercial humanoid robot *HOAP-3*, endowing it with the capacity to learn skill models from teacher demonstrations, to store them in a knowledge base, and to adapt the learned models in order to reproduce the required skills in different contexts. Different evaluation scenarios were developed to test the performance of the modules implemented in our architecture. Demonstrations were organized over two major scenarios to provide separate validation for the knowledge base system and the complete developed architecture.

Future work will constantly focus on augmenting the architecture cognitive capacities to generate better, more intelligent behaviours. By choosing to start from a bottom level definition of intelligence many assumptions and simplifications are made; this limits the possible scope of performance for the robots while reducing the complexity of the systems. These issues must be handled and solved in future work as we continue to improve the system and make it capable of performing ever more complex behaviours.

The skill learning module provides effective means for teaching the robot the desired skills. However, the teaching process is not as smooth and streamlined as it could aspire to be, and a certain level of practice and familiarity with the robot platform is required from the teacher in order to be efficient at providing demonstrations. Future work must concentrate on topics of human-robot interaction to improve the demonstration approach. The skill knowledge module affords the robot mechanisms by which to select skills to reproduce in different contexts. The implemented system is capable of performing under the demonstrated scenarios. However, these demonstrations are still limited in terms of the number of possible choices and situations they have to handle. Future work must provide comprehensive evaluations of capabilities and limitations of the skill knowledge module in a larger range of scenarios. The skill adaptation module proves functional for the requirements under the designed demonstrated scenarios. However, the module in its current implementation requires supervision from the operating user. Future work must always increase the degree of autonomy for the overall system. Also, future work would benefit from testing and user evaluations employing different users with varying levels of expertise. The implemented skill reproduction module allows a satisfactory control of the robot performance in reproducing various task. Further work is required to enhance the performance of the robot reproductions, particularly for improving execution speed and providing more natural, human-like, movements.

Acknowledgments The research leading to these results has received funding from the ROBOHEALTH-A project DPI2013-47944-C4-1-R, funded by the Spanish Ministry of Economy and Competitiveness, and from the RoboCity2030-III-CM project (S2013/MIT-2748), funded by Programas de Actividades I+D en la Comunidad de Madrid and cofunded by Structural Funds of the EU.

References

1. Albus, J.: Outline for a theory of intelligence. IEEE Trans. Syst. Man Cybern. **21**(3), 473–509 (1991). doi:10.1109/21.97471
2. Albus, J.S.: The nist real-time control system (rcs): an approach to intelligent systems research. J. Exp. Theor. Artif. Intell. **9**(2-3), 157–174 (1997)

3. Albus, J.S., Barbera, A.J.: RCS: A cognitive architecture for intelligent multi-agent systems. Ann. Rev. Control **29**(1), 87–99 (2005)
4. Anderson, J.R., Bothell, D., Byrne, M.D., Douglass, S., Lebiere, C., Qin, Y.: An integrated theory of the mind. Psychol. Rev. **111**, 1036–1060 (2004)
5. Bechtel, W.: Representations and cognitive explanations: Assessing the dynamicist's challenge in cognitive science. Cognit. Sci. **22**(3), 295–318 (1998)
6. Billard, A., Calinon, S., Dillmann, R., Schaal, S.: Robot programming by demonstration. In: Siciliano, B., Khatib, O. (eds.) Handbook of Robotics, pp. 1371–1394. Springer, Secaucus, NJ, USA (2008)
7. Brooks, R., Breazeal, C., Marjanović, M., Scassellati, B., Williamson, M.: The Cog Project: Building a Humanoid Robot. In: Computation for Metaphors, Analogy, and Agents, pp. 52–87. Springer-Verlag (1999)
8. Burghart, C., Mikut, R., Stiefelhagen, R.: A cognitive architecture for a humanoid robot: a first approach. Humanoid Robots, 357–362 (2005)
9. Calinon, S.: Robot programming by demonstration: a probabilistic approach. EPFL/CRC Press (2009)
10. Chiaverini, S., Siciliano, B.: The unit quaternion: a useful tool for inverse kinematics of robot manipulators. Syst. Anal. Model. Simul. **35**(1), 45–60 (1999)
11. Choi, D., Kang, Y., Lim, H.: Knowledge-based control of a humanoid robot. Intelligent Robots and, 3949–3954 (2009)
12. Clark, A.: Mind and causality. Advances in Consciousness Research, chap. Embodiment and the Philosophy of Mind. John Benjamins Public (2004)
13. Duch, W., Oentaryo, R.J., Pasquier, M.: Cognitive Architectures: Where Do We Go from Here? In: Proceedings of the 2008 Conference on Artificial General Intelligence 2008: Proceedings of the First AGI Conference, pp. 122–136. IOS Press, Amsterdam, The Netherlands (2008)
14. Galindo, C., Gonzalez, J., Fernández-Madrigal, J.: An Architecture for Cognitive Human-Robot Integration. Application to Rehabilitation Robotics. In: 2005 IEEE International Conference Mechatronics and Automation, vol. 1, pp. 329–334. IEEE (2005)
15. García, D.H., Monje, C.A., Balaguer, C.: Adaptation of robot skills models to new task contraints. Int. J. Humanoid Robot. **12**(3), 15500,241,155002,416 (2015). doi:10.1142/S0219843615500243
16. Hernández García, D., Monje, C.A., Balaguer, C.: Framework for Learning and Adaptation of Humanoid Robot Skills to Task Constraints. In: Armada, M.A., Sanfeliu, A., Ferre, M. (eds.) ROBOT2013: First Iberian Robotics Conference, Advances in Intelligent Systems and Computing, vol. 252, pp. 557–572. Springer International Publishing (2014)
17. Hernández García, D., Monje, C.A., Balaguer, C.: Generation and Adaptation of Robot Skills Models. In: 2014 14Th IEEE-RAS International Conference On Humanoid Robots (Humanoids), pp. 173–178 (2014)
18. Hernández García, D., Monje, C.A., Balaguer, C.: Knowledge Base Representation for Humanoid Robot Skills. In: Proceedings of the 19Th IFAC World Congress, 2014, vol. 19, pp. 3042–3047. International Federation of Automatic Control (2014)
19. Ijspeert, A., Nakanishi, J., Schaal, S.: Trajectory formation for imitation with nonlinear dynamical systems. In: 2001 IEEE/RSJ International Conference on Proceedings Intelligent Robots and Systems, vol. 2, pp. 752–757. doi:10.1109/IROS.2001.976259
20. Jung, Y., Choi, Y., Park, H., Shin, W., Myaeng, S.H.: Integrating Robot Task Scripts with a Cognitive Architecture for Cognitive Human-Robot Interactions, 152–157 (2007). doi:10.1109/IRI.2007.4296613
21. Kajita, S., Kanehiro, F., Kaneko, K., Yokoi, K., Hirukawa, H.: The 3D Linear Inverted Pendulum Mode: a Simple Modeling for a Biped Walking Pattern Generation. In: Proceedings of IEEE/RSJ International Conference On Intelligent Robots and Systems, vol. 1, pp. 239–246 (2001)
22. Khansari-Zadeh, S., Billard, A.: Imitation learning of globally stable non-linear point-to-point robot motions using nonlinear programming. In: 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2676–2683 (2010). doi:10.1109/IROS.2010.5651259
23. Khansari-Zadeh, S., Billard, A.: Learning stable nonlinear dynamical systems with gaussian mixture models. IEEE Trans. Robot. **27**(5), 943 –957 (2011). doi:10.1109/TRO.2011.2159412
24. Kieras, D.E., Meyer, D.E.: An overview of the epic architecture for cognition and performance with application to human-computer interaction. Hum.-Comput. Interact. **12**(4), 391–438 (1997)
25. Kim, K., Lee, J.Y., Choi, D., Park, J.M., You, B.J.: Autonomous task execution of a humanoid robot using a cognitive model. In: 2010 IEEE International Conference on Robotics and Biomimetics, pp. 405–410 (2010). doi:10.1109/ROBIO.2010.5723361
26. Krüger, N., Piater, J., Wörgötter, F., Geib, C., Petrick, R., Steedman, M., Ude, A., Asfour, T., Kraft, D., Omrcen, D., Hommel, B., Agostino, A., Kragic, D., Eklundh, J., Krüger, V., Dillmann, R.: A Formal Definition of Object Action Complexes and Examples at Different Levels of the Process Hierarchy. Technical Report, EU project PACO-PLUS (2009)
27. Laird, J.E., Newell, A., Rosenbloom, P.S.: Soar: an architecture for general intelligence. Artif. Intell. **33**(1), 1–64 (1987)
28. Langley, P., Laird, J.E., Rogers, S.: Cognitive architectures: Research issues and challenges. Cogn. Syst. Res. **10**(2), 141–160 (2009). doi:10.1016/j.cogsys.2006.07.004
29. Lemaignan, S., Ros, R., Mo senlechner, L., Alami, R., Beetz, M.: ORO, a knowledge management platform for cognitive architectures in robotics. In: 2010 IEEE/RSJ International Conference On Intelligent Robots and Systems (IROS), pp. 3548–3553. IEEE (2010)
30. Levesque, H., Lakemeyer, G.: Chapter 23 cognitive robotics. In: Frank van Harmelen, V.L., Porter, B. (eds.) Handbook of Knowledge Representation, vol. 6526, pp. 869–886. Elsevier (2008). doi:10.1016/S1574-6526(07)03023-4
31. Monje, C.A., Pierro, P., Balaguer, C.: Humanoid robot rh-1 for collaborative tasks: a control architecture for human-robot cooperation. Appl. Bionics Biomech. **5**(4), 225–234 (2008)
32. Pierro, P., Hernandez, D., Gonzalez-Fierro, M., Blasi, L., Milani, A., Balaguer, C.: Humanoid Teleoperation System

for Space Environments. In: International Conference On Advanced Robotics, 2009. ICAR 2009, pp. 1 –6 (2009)
33. Poole, D., Mackworth, A., Goebel, R.: Computational intelligence: a logical approach. Oxford university Press, USA (1998)
34. Scassellati, B.: Theory of mind for a humanoid robot. Auton. Robot. **12**(1), 13–24 (2002). doi:10.1023/A:1013298507114
35. Schaal, S.: Is imitation learning the route to humanoid robots? Trends Cogn. Sci. **6**, 233–242 (1999)
36. Shanahan, M.: A cognitive architecture that combines internal simulation with a global workspace. Conscious. Cogn. **15**, 433–449 (2006). doi:10.1016/j.concog.2005.11.005
37. Siciliano, B., Sciavicco, L., Villani, L., Oriolo, G.: Robotics: Modelling, planning and control, 2nd edn. Advanced Textbooks in Control and Signal Processing. Springer (2009)
38. Sun, R.: Multi-agent systems for society. chapter Cognitive Architectures and Multi-agent Social Simulation, pp. 7–21. Springer-Verlag, Berlin, Heidelberg (2009)
39. Tan, H.: The future of humanoid robots - research and applications, chapter Implementation of a Framework for Imitation Learning on a Humanoid Robot Using a Cognitive Architecture, pp. 191–210. InTech (2012)
40. Tenorth, M., Beetz, M.: Knowrob: a knowledge processing infrastructure for cognition-enabled robots. Int. J. Robot. Res. (IJRR) **32**(5), 566–590 (2013)
41. Vernon, D., Metta, G., Sandini, G.: A survey of artificial cognitive systems: Implications for the autonomous development of mental capabilities in computational agents. IEEE Trans. Evol. Comput. **11**(2), 151–180 (2007). doi:10.1109/TEVC.2006.890274
42. Waibel, M., Beetz, M., Civera, J., D'Andrea, R., Elfring, J., Galvez-Lopez, D., Haussermann, K., Janssen, R., Montiel, J.M.M., Perzylo, A., Schiessle, B., Tenorth, M., Zweigle, O., Van de Molengraft, R.: Roboearth. IEEE Robot. Autom. Mag. **18**(2), 69–82 (2011). doi:10.1109/MRA.2011.941632
43. Zoliner, R., Pardowitz, M., Knoop, S., Dillmann, R.: Towards Cognitive Robots: Building Hierarchical Task Representations of Manipulations from Human Demonstration. In: Proceedings of the 2005 IEEE International Conference On Robotics and Automation, 2005. ICRA 2005, pp. 1535–1540. IEEE (2005)

Daniel Hernández García received his M.S. degree in Robotics and Automation and Ph.D. degree in Electrical Engineering, Electronics and Automation from the Universidad Carlos III of Madrid, in 2010 and 2014, respectively. He has been a Researcher at the Universidad Carlos III of Madrid, and worked on various European and national projects. His research interests include Robot Control, in particular Humanoid Robotics, Human-Robot Interaction, Machine Learning Algorithms and Cognitive Systems.

Concepción A. Monje received her MSc. Degree in Electronics Engineering from the Industrial Engineering School of the University of Extremadura, Spain, in 2001, and her Ph.D. Degree in Industrial Engineering from the University of Extremadura, Spain, in 2006. In September 2006 she joined the University Carlos III of Madrid as a Visiting Professor in the Department of Electronics and Electromechanical Engineering. Her research focuses on control theory and applications of fractional calculus in control systems and robotics. She has been working in these areas for over 8 years, and she has published over 50 technical papers mostly related to control and robotics.

Carlos Balaguer received his Ph.D. in Automation from the Polytechnic University of Madrid (UPM), Spain, in 1983. From 1983–1994 he was with the Department of Systems Engineering and Automation of the UPM as Associated Professor. Since 1996, he has been a Full Professor of the RoboticsLab at the University Carlos III of Madrid. His research has included humanoid and assistive and service robots, among others. He participates in numerous EU projects since 1989, such as Eureka projects SAMCA, AMR and GEO, Esprit projects ROCCO and CEROS, Brite project FutureHome, IST project MATS, and the 6FP IP projects ManuBuild, I3CON, and Strep Robot@CWE. He has published more than 180 papers in journals and conference proceedings, and several books in the field of robotics.