Control of complex actions in humans and robots

Kleijn, R.E. de

Author: Kleijn, Roy de
Title: Control of complex actions in humans and robots
Date: 2017-11-23

CHAPTER 3

Robotics and human action

THE FIELD OF ROBOTICS is shifting from building industrial robots that can perform repetitive tasks accurately and predictably in constrained settings, to more autonomous robots that should be able to perform a wider range of tasks, including everyday household activities. To build systems that can handle the uncertainty of the real world, it is important for roboticists to look at how humans are able to perform in such a wide range of situations and contexts—a domain that is traditionally the purview of cognitive psychology. Cognitive scientists have been rather successful in bringing computational systems closer to human performance. Examples include image and speech recognition and general knowledge representation using parallel distributed processing (e.g. modern deep learning models).

Similarly, cognitive psychologists can use robotics to complement their research. Robotic implementations of cognitive systems can act as a "computational proving ground", allowing accurate and repeatable real-world testing of model predictions. All too often, theoretical predict-

This chapter is an adaptation of the book chapter *de Kleijn, R., Kachergis, G., & Hommel, B. (2015). Robotic action control: On the crossroads of cognitive psychology and robotics. In H. Samani (Ed.),* Cognitive robotics. *Taylor & Francis.*

ions—and even carefully conducted model simulations—do not scale up or even correspond well to the complexity of the real world. Psychology should always seek to push theory out of the nest of the laboratory and see if it can take flight. Finally, cognitive psychologists have an opportunity to conduct experiments that will both inform roboticists as they seek to make more capable cognitive robots, and increase our knowledge of how humans perform adaptively in a complex, dynamic world. In this chapter, we will give a broad but brief overview of the fields of cognitive psychology and robotics, with an eye to how they have come together to inform us about how (artificial and natural) actions are controlled.

3.2 Early history of the fields

3.2.1 History of cognitive psychology

Before cognitive psychology and robotics blended into the approach now known as cognitive robotics, both fields already had a rich history. Cognitive psychology as we now know it has had a rocky past (as have most psychological disciplines, for that matter). Breaking away from philosophy, after briefly attempting to use introspection to observe the workings of the mind, the field of psychology found it more reliable to rely on empirical evidence.

Although making rapid strides using this empirical evidence, for example in the form of Donders' now classic reaction time experiments which proposed stages of processing extending from perception to action, early cognitive psychology came to be dominated by a particular approach, *behaviorism*. This position, popularized by Watson [169] and pushed further by Skinner [143], held that the path for psychology to establish itself as a natural science on par with physics and chemistry would be to restrict itself to observable entities such as stimuli and responses. In this sense, behaviorists such as Skinner were strongly antirepresentational, i.e. against the assumption of internal knowledge and states in the explanation of behavioral observations. On the other hand, the focus on observable data brought further rigor into the field, and many interesting effects were described and explained.

The behaviorist approach dominated the field of psychology during the first half of the 20th century. In the 1950s, seeming limitations of behaviorism fueled what some scholars would call the *neocognitive revolution*. Starting with Chomsky's scathing 1959 review of Skinner's book [27] that tried to explain how infants learn language by simple association, many researchers were convinced that behaviorism could not explain fundamental cognitive processes such as learning (especially language) and memory. The foundations of the field of artificial intelligence were also nascent, and pursuing explanations of high-level, uniquely human aptitudes—e.g. analytical thought, reasoning, logic, strategic decision-making—grew in popularity.

3.2.2 The computer analogy

Another factor contributing to the neocognitive revolution was the emergence of a new way to describe human cognition as similar to electronic computer systems. The basic mechanism operating computers was (and still is, in a fundamental way) gathering input, processing it, and outputting the processed information, not unlike the basic cognitive model of stimulus detection, storage and transformation of stimuli, and response production.

Clearly, this processing of information requires some representational states which are unaccounted for (and dismissed as unnecessary) by behaviorists. This new way to look at human cognition as an information processing system not only excited psychologists as a way of understanding the brain, but the analogy also raised hopes for building intelligent machines. The idea was that if computer systems could use the same rules and mechanisms as the human brain, they could also *act* like humans. Perhaps the most well-known proponent of this optimistic vision was Turing [161], who suggested that it wouldn't be long before machine communication would be indistinguishable from human communication. Maybe the secret of cognition lies in the way the brain gathers, stores, and subsequently manipulates data, it was thought.

Alas, the optimists would be disappointed. It soon became clear that computers and humans have very different strengths and weaknesses. Computers can calculate half a million decimals of π within a second. Humans can read terrible handwriting. Clearly, humans are not so comparable to basic input–output systems after all. It would take another 25 years for cognitive psychology and artificial intelligence to begin their romance once again, in the form of the *parallel distributed processing* (PDP) approach [130].

3.2.3 Early cognitive robots

With this idea of smart computer systems in mind, it seemed almost straightforward to add embodiment to build intelligent agents. The first cognitive robots were quite simple machines. The *Machina Speculatrix* [166] consisted of a mobile platform, two sensors, actuators and "nerve cells". Understandably, these robots were designed to mimic behavior of simple animals, and could move safely around a room and recharge themselves using relatively simple approach and avoidance rules. Due to their simplicity, it was questionable exactly how *cognitive* these robots were—they are more related to cybernetics and control theory (e.g. [8])—but soon enough complexity made its way into cognitive robotics.

From the 1960s, robots would be able to represent knowledge and plan sequences of operations using algorithms such as STRIPS [47], that would now be considered essential knowledge for every AI student. The STRIPS planner, which represents goal states and preconditions and attempts to derive the action sequences that would achieve them before carrying them out, is quite slow to execute. Moreover, this type of planning suffers from its closed world assumption (i.e. that the environment and all relevant states are known—by programming—and will not change), and the massive complexity of the real world, leading to intractable computations. Yet the general approach taken by STRIPS—of modeling the environment, possible actions and state transformations, and goal states via predicate logic, and operating robots via a sense-plan-act loop—has dominated cognitive robotics for quite some time, and is still a strong thread today.

Various behavior-based robotics architectures and algorithms—taking some inspiration from biological organisms—have been developed in the past few decades. An early, influential example is Rodney Brooks' *subsumption architecture* [21], which eschews planning entirely; "planning is just a way of avoiding to figure out what to do next", using a defined library of basic behaviors arranged hierarchically to generate behavior based on incoming stimuli. Although fast and often generating surprisingly complex behavior from simple rules (see also [20]), the subsumption architecture and many other behavior-based robotics algorithms do not yet incorporate much from the lessons to be learned from psychological studies in humans.

3.3 Action control

3.3.1 Introduction

One of the other areas that shows considerable overlap between robots and humans is motor or action control. Two types of control systems govern motor action: *feedforward* and *feedback* control systems.

A feedforward motor control system sends a signal from the (human or robotic) motor planning component to the relevant motor component using predetermined parameters, executing said action. Information from the environment can be considered only before execution begins, which makes feedforward control suitable for predictable environments. In contrast, a feedback motor control system incorporates information from itself or the environment (feedback) more or less continuously to modulate the control signal. In this way, the system can dynamically alter its behavior in response to a changing environment.

3.3.2 Feedforward and feedback control in humans

For many years, psychology and related disciplines have approached action control from rather isolated perspectives. As the probably first systematic study on movement control by Woodworth [171] had provided strong evidence for the contribution of environmental information, many authors have tried to develop closed-loop models of action control that rely on a continuous feedback loop (e.g. [1]). At the same time, there was strong evidence from animal and lesion studies [81, 155] and from theoretical considerations [87] that various movements can be considered in the absence of sensorimotor feedback loops, which has motivated the development of feedforward models (e.g. [59]).

Schmidt [140] was one of the first who argued that human action control consists of both feedforward and feedback components. According to his reasoning, human agents prepare a movement schema that specifies the relevant attributes of the intended movement but leave open parameter slots that are specified by using online environmental information. In particular, feedforward mechanisms seem to determine the necessary action components offline and pre-load at least some of them before initiating the action [59], and to selectively tune attention to stimuli and stimulus dimensions that are relevant to the task [64]. Feedback processes, in turn, provide excellent accuracy—often at the cost of speed [141]. These strengths and weaknesses have motivated hybrid models claiming that feedforward mechanisms provide the skeleton of action plans which leave open slots for parameters provided by feedback processes. Neuroscientific evidence has provided strong support for such a hybrid control model, suggesting that offline action planning along a ventral cortical route is integrated with online sensorimotor specification along a dorsal route [53, 54, 64, 140].

A particularly good example of this kind of interaction is provided by the observations of Goodale et al. [55]. In a clever experiment, participants were asked to rest their hand on a platform and point to a visual target presented at a random location on an imaginary line in their right visual field. The participants were not told that in half of the trials the target would change location during the first saccade. The authors found that participants would successfully point to the target on these trials without even being aware of the location change, and without additional delay. As feedforward programming is assumed to take time, a fast and online feedback mechanism of which participants are unaware has to be responsible for this finding.

On a higher level, interaction between feedforward and feedback systems must exist for goal-directed action to be carried out. Higher-level, goal-directed action planning, such as planning to make pancakes would be impossible to plan in a completely feedforward fashion: it would require all motor parameters to be specified *a priori*, and thus would require exact knowledge of the position and properties of all necessary equipment and ingredients, such as weight, friction coefficients, et cetera.

Instead, many of these parameters can be filled in online by using information from the environment. It is not necessary to know the exact weight of a pan, because you can determine that easily by picking it up: you increase the exerted force until the pan leaves the surface of the kitchen counter. This does not rule out a complementary role for feedforward parameter estimation: you likely also learn a distribution of probable pan weights (e.g. more than 50 g and less than 10 kg) from your experience of other pans—or even just similarly-sized objects.

Interaction between feedforward and feedback becomes even more apparent on a higher level when a planned action fails to be executed. When a necessary ingredient is missing, replanning (or cancellation) of a preprogrammed action sequence may be necessary: if there is no butter, can I use oil to grease up the pan? Somehow, this information gathered by feedback processes must be communicated to the higher level action planner.

3.3.3 Feedforward and feedback control in robots

The theorizing on action control in robotic systems must be considered rather ideological, sometimes driven by the specifics of particular robots or tasks considered and sometimes by broadly generalized antirepresentationalist attitudes. Many early robots only had a handful of sensors and responded in a fixed pattern of behavior given a particular set of stimuli. Some robots were even purely feedforward, performing the same action or action sequence, with no sensory input whatsoever [106]. Feedforward or simple reactive control architectures make for very brittle behavior: even complex, carefully-crafted sequences of actions and reactions will appear clumsy if the environment suddenly presents an even slightly novel situation.

More complex architectures have been proposed, often with some analogy to biology or human or animal behavior, giving birth to the field of *behavior-based robotics.* The *subsumption architecture* [21] was a response to the traditional GOFAI, and posited that complex behavior need not necessarily require a complex control system. Different behaviors are represented as layers that can be inhibited by other layers. For example, a simple robot could be provided with the behaviors *wandering, avoiding, pickup,* and *homing.* These behaviors are hierarchically structured, with each behavior inhibiting its preceding behavior [7].

This hierarchy of inhibition between behavior is (although somewhat more complex) also visible in humans. For example, if your pants are (accidentally) set on fire while doing the dishes, few people would finish the dishes before stopping, dropping, and rolling. In other words, some behaviors take precedence over others. An approach similar to the subsumption architecture has been proposed by Arkin [6]. The *motor schema* approach also uses different, parallel layers of behavior, but does not have the hierarchical coordination that the subsumption approach does. Instead, each behavior contributes to the robot's overall response.

On a higher level, as noted in the previous section, other problems arise. When a planned action fails to succeed, for example because a robot can't find a pan to make pancakes in, replanning is necessary. The earliest AI planners such as GPS would simply backtrack to the previous choice point and try an alternative subaction. However, this does not guarantee the eventual successful completion of the action. Other planners, such as ABSTRIPS [134], use a hierarchy of representational levels. When it fails to complete a subaction, it could return to a more abstract level.

However, truly intelligent systems should be more flexible in handling such unforeseen events. If a robot cannot make me a pizza with ham, maybe it should make me one with bacon? Generalization and substitution remain an elusive ability for robots, although vector space models of semantics (e.g. BEAGLE [72]) offer a step in the right direction. Like neural networks, these models represent items (e.g. words) in a distributed fashion, using many-featured vectors with initially low similarity between random items. As the model learns—say, by reading documents—item representations are updated to make them more similar (on a continuous scale) to contextually similar items. These continually-updated representations can be used to extract semantic as well as syntagmatic (e.g. part-of-speech) relationships between items. Beyond text learning, vector space models may ultimately be used to learn generalizable representations for physical properties and manipulations of objects and environments.

3.3.4 Robotic action planning

It is understood that reaching movements in humans have an initial ballistic feedforward component, followed by a slower feedback-driven component that corrects for error in the initial movement. As people become more adept at reaching to targets at particular distances, a greater portion of their movement is devoted to the initial feedforward component and less time is spent in the feedback component, thus speeding response times. Understanding how this happens should enable roboticists to make more adaptive, human-like motor planning systems for robots.

In this line of research, Kachergis et al. [75] studied sequence learning using mouse movements. Inspired by earlier work of Nissen and Bullemer [107], subsequences of longer sequences were acquired by human participants during a learning phase. The participants seem to implicitly extract the subsequences from longer sequences by showing faster response times and context effects.

These findings cast doubt on a simple chaining theory of sequential action. Rosenbaum et al. [129] interpreted these findings as evidence that sensory feedback is not a necessary component for action sequencing, in keeping with the conclusion of Lashley [87]. They argued that "the state of the nervous system can predispose the actor to behave in particular ways in the future," (p. 526), or, there are action plans for some behaviors. And yet, studies on spontaneous speech repair (e.g. [103]) also show that people are very fast in fixing errors in early components of a word or sentence, much too fast to assume that action outcomes are evaluated only after entire sequences are completed. This means that action planning cannot be exclusively feedforward, as Lashley [87] seemed to suggest, but must include several layers of processing, with lower levels continuously checking whether the current action component proceeds as expected. In other words, action planning must be a temporally extended process in which abstract representations to some extent provide abstract goal descriptions, which must be integrated with lower-level subsymbolic representations controlling sensorimotor loops. The existence of subsymbolic sensorimotor representations would account for context and anticipation effects, as described above.

The main lesson for robotic motor planning is that purely symbolic planning may be too crude and context-insensitive to allow for smooth and efficient multi-component actions. Introducing multiple levels of action planning and action control may complicate the engineering considerably, but it is also likely to make robot action more flexible and robust—and less "robotic" to the eye of the user.

3.4 Acquisition of action control

3.4.1 Introduction

In order for humans or robots to be able to achieve their goals, it is necessary for them to know what effect an action would have on their environment. Or, reasoning back to the inverse model, what actions are required to produce a certain effect in the environment. Learning relevant action–effect bindings as an infant is a fundamental part of development and likely bootstraps later acquisition of general knowledge.

In humans, learned action–effects seem to be stored bidirectionally. Following Lotze [91] and Harless [57], James [69] noted that intentionally creating a desired effect requires knowledge about, and thus the previous acquisition of action–effect contingencies. The *Theory of Event Coding* (TEC) is a comprehensive empirically well-supported theoretical framework explaining the acquisition and use of such action–effect bindings for goal-directed action ([65], for recent reviews see [63, 142]). TEC states that actions and their expected effects share a common neural representation. Therefore, performing an action activates the expectation of relevant effects and thinking of (i.e. intending or anticipating) an action's effects activates motor neurons responsible for achieving those effects.

3.4.2 Human action–effect learning

In traditional cognitive psychology experiments, action–effect bindings are acquired by having humans repetitively perform an action (such as pressing a specific button on a keyboard), after which an effect (such as a sound or a visual stimulus) is presented. After a certain amount of exposure to this combination of action and effect, evidence suggests that a bidirectional binding has been formed. When primed with a previously learned effect, people respond faster with the associated action [42]. This action–effect learning is quite robust but sensitive to action–effect contingency and contiguity [43].

Of course, action–effect learning does not only happen in artificial environments such as psychology labs. In fact, action–effect learning in humans starts almost instantly after birth [164] and some would argue even before. Young infants perform uncoordinated movements known as *body* or *motor babbling*. Most of these movements will turn out to be useless. However, some of them will have an effect that provides the infant with positive feedback. For example, a baby could accidentally push down with its right arm while lying on its belly, resulting in rolling on its back and seeing all sorts of interesting things. Over time, the infant will build up action–effect associations for actions it deems useful, and can perform motor acts by imagining their intended effects.

Having mastered the intricacies of controlling the own body, higher level action–effects can be learned in a manner similar to motor babbling. Eenshuistra et al. [39] give the example of piloting a spacecraft that you are trying to slow down. If nobody ever instructed you on how to do that, your best option would probably be pressing random buttons until the desired effect is reached (be careful with that self-destruct button!). Once you have learned this action–effect binding, performance in a similar situation in the future will be much better.

3.4.3 Robotic action–effect learning

The possibility that cognition can be grounded in sensorimotor experience and represented by automatically created action–effect bindings has attracted some interest of cognitive roboticists already. For instance, Kraft et al. [82] have suggested a three-level cognitive architecture that relies on object-action complexes, that is, sensorimotor units on which higher-level cognition is based. Indeed, action–effect learning might provide the cognitive machinery to generate action-guiding predictions and the offline, feedforward component of action control. This component might specify the invariant aspects of an action, that is, those characteristics that need to be given for an action to reach its goal, to create its intended effect while an online component might provide fresh environmental information to specify the less goal-relevant parameters, such as the speed of a reaching movement when taking a sip of water from a bottle [64]. Arguably, such a system would have the benefit of allowing for more interesting cognitive achievements than the purely online, feedback-driven systems that are motivated by the situated-cognition approach [22]. At the same time, it would be more flexible than systems that rely entirely on the use of internal forward models [36]. Thus, instead of programmers trying to imagine all possible scenarios and enumerate reasonable responses, it might be easier to create robots that can learn action–effect associations appropriate to their environment and combine them with online information.

In robots as well as in humans, knowledge about one's own body is required to acquire knowledge about the external world. Learning how to control your limbs—first separately and then jointly (e.g. walking)—clearly takes more than even the first few years of life: after learning to roll over, crawl, and then walk, we are still clumsy at running and sport for several years (if, indeed, we ever become very proficient). Motor babbling helps develop tactile perception and proprioception—as well as visual and even auditory cues—of what our body in motion feels like. Knowing these basic actions and their effects on ourselves (e.g. what hurts) lays the foundation for learning how our actions can affect our environments.

In perhaps the first ever study of motor babbling in a (virtual) robot, Kuperstein [84] showed how random movement execution can form associations between a perceived object-in-hand position and the corresponding arm posture. This association is bidirectional, and as such is in line with ideomotor (or TEC) theory. We (and others, e.g. [25]) believe that such bidirectional bindings can help robots overcome traditional problems, such as inverse model inference from a forward model.

More recent investigations in robotic motor babbling have extended and optimized the method to include behavior that we would consider *curiosity* in humans. For example, Saegusa et al. [135] robotically implemented a sensorimotor learning algorithm that organized learning in two phases: *exploration* and *learning*. In the exploration stage, random movements are produced, while in the learning stage the action–effect bindings (or, more specifically, mapping functions) are optimized. The robot can then direct more effort to learning bindings that have not yet been learned well.

3.5 Directions for the future

3.5.1 What's next?

Many questions remain with respect to the acquisition and skillful performance of not only well-specified, simple actions (e.g. reaching to a target) but of complex actions consisting of various components and involving various effectors. Indeed, how can we create a learning algorithm that can go from basic motor babbling to both successful goal-directed reaching, grasping, and manipulation of objects? To accomplish this obviously difficult goal, it will likely be beneficial for psychologists to study infants' development of these abilities and beneficial for cognitive roboticists to learn more from human capabilities.

3.5.2 Affordance learning

Object manipulation and use is an indispensable activity for robots working in human environments. Perceiving object affordances—i.e. what a tool can do for you or how you can use an object—seems to be a quick, effortless judgment for humans, in many cases. For example, when walking around and seeing a door, you automatically pull the handle to open it.

One of the ways robots can perform object affordance learning is by motor babbling using simple objects as manipulators (e.g. [152]). In a so-called *behavioral babbling stage* a robot applies randomly chosen behaviors to a tool and observes their effects on an object in the environment. Over time, knowledge about the functionality of a tool is acquired, and can be used to manipulate a novel object with the tool.

As impressive as this may sound, this approach does not allow for easy generalization, and the robot cannot use this knowledge to manipulate objects using another, similar, tool. More recent approaches, such as demonstrated by Jain and Inamura [68] infer functional features from objects to generalize affordances to unknown objects. These functional features are supposed to be object invariant within a tool category.

In humans, an approach that seems successful in explaining affordance inference is based on Biederman's *recognition-by-components theory* [15]. This theory allows for object recognition by segmenting an encountered object in elementary geometric parts called *geons*. These are simple geometric shapes such as cones, cylinders and blocks. By reducing objects to a combination of more elementary units invariance is increased, simplifying object classification. Biederman recognized 36 independent geons, having a (restricted) generative power of 154 million three-geon objects.

In addition to being useful for object classification, geons can also be used to infer affordances. For example, a spoon is suitable for scooping because its truncated hollow sphere at the end of its long cylinder allows for containing things, and an elongated cylinder attached to an object can be used to pick it up. One very promising example of the use of geons in affordance inference is demonstrated by Tenorth and Beetz [157]. This technique matches perceived objects to three-dimensional CAD models from a public database such as Google Warehouse. These models are then segmented into geons, which makes affordance inference possible.

However, the affordances that geons give us need to be learned in some way. Teaching robots how to infer what a tool can be capable of remains difficult. Ultimately, we want affordances to develop naturally during learning: be it from watching others, from verbal instruction, or from embodied experimentation. Task context is also an important aspect of affordance learning: depending on the situation, a hammer can be used as a lever, a paperweight, a missile, or well, a hammer. To understand how context affects action planning, studying naturalistic scenes and human activities jointly seems essential (cf. [2]).

Learning geon affordances that can be generalized to object affordances seems a fruitful approach to automating affordance learning in robots, although it is early to say whether this or other recent approaches will fare better. For example, deep neural networks use their multiple hidden layers along with techniques to avoid overfitting to learn high-level perceptual features for discriminating objects. The representations learned by such networks are somewhat more biologically plausible than geon decompositions, and thus may be more suitable for generalization (although cf. [154] for recently discovered generalization problems with deep neural networks).

3.5.3 Everyday action planning

A major obstacle in the way of robots performing everyday actions is the translation of high-level, symbolic task descriptions into sensorimotor action plans. In order to make such translations, one method would be to learn the other way around: by observing sensorimotor actions, segment and classify the input.

Everyday action is characterized by sequential, hierarchical action subsequences. Coffee and tea-making tasks, for example, have shared subsequences such as adding milk or sugar. Moreover, the goal of adding sugar might be accomplished in one of several ways: e.g. tearing open and adding from a packet, or spooning from a bowl or box. Also, these subsequences do not necessarily have to be performed in the same order every time (with some constraints, of course). It is this flexibility and ability to improvise that makes everyday action so natural for humans, yet so hard for robots.

Cognitive models that represent hierarchical information have been proposed (e.g. [18, 33]), but differ in the way they represent these hierarchies. One approach explicitly represents action hierarchies by hardcoding them into the model—hardly something we can do for a general autonomous robot—whereas the latter models hierarchy as an emergent property of the recurrent neural network. More recently, the model put forth by Kachergis et al. [76], uses a recurrent neural network with biologically plausible learning rules to extract hierarchies from observed sequences, needing far fewer exemplars than previous models.

3.6 Conclusion

In this chapter, we have discussed several concepts that are shared between cognitive robotics and cognitive psychology in order to argue that the creation of flexible, truly autonomous robots depends on the implementation of algorithms that are designed to mimic human learning and planning. Thus, there are many relevant lessons from cognitive psychology for aspiring cognitive roboticists.

Ideomotor theory and its implementations such as TEC provide elegant solutions to action–effect learning. Robotic motor learning algorithms that use motor babbling to bootstrap higher-order learning seem to be promising, and require little *a priori* knowledge given by the programmer, ultimately leading to more flexible robots.

Generalization of action plans is still a very difficult problem. Inferring hierarchical structure of observed or learned action sequences seems to be a promising approach, although the structure of everyday action appears to be nearly as nuanced and intricate to untangle as the structure of human natural language—and less well-studied, at this point. Again, we believe that biologically inspired learning models such as LeabraTI can play a role in making robotic action more human-like.

The overlapping interests of cognitive robotics and cognitive psychology have proven fruitful so far. Mechanisms like motor babbling and affordance inference, which are extensively studied in humans, can provide robots with techniques to make their behavior more flexible and human-like. We believe human inspiration for robots can be found at an even lower level by incorporating biologically-inspired neural models for learning in robots.