Title
Learning and representation of causative motor actions: a neural network model and its use in an embodied theory of language

Journal
Proceedings of the Annual Meeting of the Cognitive Science Society, 35(35)

ISSN
1069-7977

Authors
Lee-Hand, Jeremy
Knott, Alistair

Publication Date
2013

Peer reviewed

Learning and representation of causative motor actions: a neural network model and its use in an embodied theory of language

Jeremy Lee-Hand (leehandjeremy@gmail.com) and **Alistair Knott** (alik@cs.otago.ac.nz)
Department of Computer Science, University of Otago, New Zealand

Abstract

In this paper we present a neural network model of motor learning structured around circuits which associate motor actions with their sensory effects, as proposed by Hommel *et al.* (2001). The network implements a novel model of *causative actions*, which bring about specified distal movements in manipulated target objects (e.g. bending a lever). It also serves as the basis for a novel embodied account of the syntax of causative sentences such as *John bent the lever*.

Keywords: motor control, neural networks, embodied models of language

Effect-based action representations in neuroscience and language

A common idea in models of action representation is that an agent's actions (also known as motor programs) are encoded in a way which makes reference to the sensory effects they bring about. This idea has a long history, but in recent research it is most strongly associated with Prinz's (1997) theory of 'common coding' and Hommel *et al.*'s theory of 'event codes' (Hommel *et al.*, 2001). The key idea uniting these models is that motor programs are not defined purely within the motor domain: their neural representation includes a specification of the sensory effects they bring about, in one or more sensory modalities. This position can be supported both on theoretical grounds and through experiments; we will give brief examples of each kind of argument.

Theoretically, a strong argument for this view of action representation comes from considerations about how actions are learned. It is uncontroversial that an agent's repertoire of motor programs is learned through some kind of reinforcement. A reinforcing signal is a sensory signal. When an agent executes a motor program and generates a rewarding signal, an association is made between the sensory signal and this particular program. After a certain amount of training, if all goes well, the sensory signal will become associated with a range of related motor movements, which bring it about in different ways or under different circumstances, perhaps in ways which are parameterised or organised by features of the sensory stimulus. At this point, if the agent activates the sensory signal, this will bring about one of these movements, and result in reward. But equally importantly, the group of motor movements associated with the sensory signal can now be thought of as comprising an *action category*, in virtue of their shared ability to evoke the stimulus. Categories are defined around central concepts or prototypes, and in this case the unifying concept is a sensory one. For this reason, it makes sense to talk about action categories as being defined by the sensory effects they bring about.

Experimentally, the idea that actions are defined by their effects has been suported in several ways. For instance, there have been many studies exploring variations on the well-known stimulus-response compatibility effect (Simon, 1969). A good example is a study by Hommel (1993). Here subjects had to respond to an auditory stimulus by pressing a button, either with the left or right hand. The tone of the auditory stimulus indicated which button the subject should press. But as a distracting factor, the stimulus was also presented either on the left or the right. The classical stimulus-response compatibility effect is that subjects are slower to respond if the spatial location of the stimulus is incompatible with the hand which must respond. In Hommel's experiment, button presses generated a reafferent visual stimulus whose location could be decoupled from the location of the hand pressing the button, to explore whether the compatibility effect operates in the domain of motor movements or that of their sensory consequences. Button presses consistently produced a visual stimulus: illumination of a light. In one condition the light appeared on the same side as the hand (e.g. left button presses illuminated a light on the left), while in another it appeared on the opposite side (e.g. left button presses illuminated a light on the right). Hommel found that the stimulus-response compatibility effect depended on compatibility with the perceptual effects of button-presses, rather than on the hand which was used. This shows that the way subjects encode actions does make some reference to their sensory consequences—at least enough to interfere with stimulus-response mappings. Effect-based representations of motor actions are also supported by several studies of the neural representation of actions; see for instance Umiltà *et al.* (2008); Matsumoto *et al.* (2003).

Another interesting piece of evidence for effect-based action representations comes from a completely different area of cognitive science: linguistics. The evidence comes from a phenomenon called the 'causative alternation'. This is found in many languages, but we will illustrate with English. Consider the following two sentences:

(1) John bent the lever.

(2) The lever bent.

As these show, the verb *bend* can be used in a transitive sentence, where the lever appears as the object (Example 1) or in an intransitive sentence, where it appears as the subject (e.g. Example 2). On the face of it, a syntactician would have to assume two different senses of the word *bend*: one which describes the lever as an agent and one as a patient. But this is counterintuitive, since what happens to the lever is the same

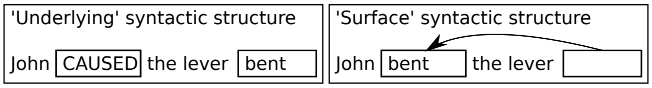
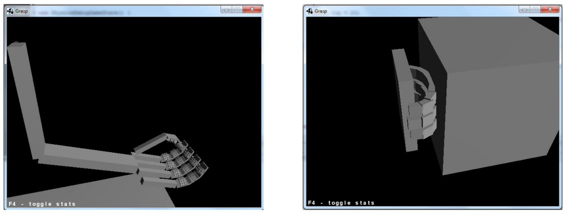


Figure 1: Derivation of *John bent the lever* by movement from an underlying syntactic structure



(a) (b)

Figure 2: (a) The hand/arm (b) detail of a single finger pad

in each case. A way to avoid assuming this implausible ambiguity in the verb *bend* is to argue that the transitive sentence *John bent the lever* really means *John caused [the lever to bend]*. This analysis can be neatly expressed in syntactic theories which posit that sentences have an 'underlying' syntactic structure which is distinct from their surface form: an idea associated with Chomskan accounts of syntax (see e.g. Chomsky, 1995). In a Chomskyan framework, we can argue that the underlying structure of Example 1 is *John caused [the lever bent]*, as shown on the left of Figure 1. At this level of analysis, 'the lever' is the subject of *bend*, just as it is in Example 2. In a Chomskyan model, the surface structure of Example 1 is produced by moving the lower verb *bent* into the position of the higher verb *caused*, as shown on the right of Figure 1.

In this paper we have two aims. We will first introduce a computational model of the learning and control of causative actions, which implements a particular take on Hommel *et al.*'s conception of event codes. The model has several interesting features as an account of action representation, which we will briefly discuss. But our other main aim is to juxtapose an account of processing in the motor control network with the syntactic analysis of causative verbs just sketched above. We will argue that the network may provide a framework that allows the syntactic analysis to be expressed in terms of neural mechanisms.

A platform for learning and control of simulated actions

Our computational model was implemented in a software environment for simulating hand/arm actions called GraspProject (Lee-Hand *et al.*, 2012; for details see Neumegen, 2013). The environment is built on top of the JMonkey games engine, which uses the Bullet physics engine to define objects made up of linked rigid bodies, and OpenGL to render graphical views of these. GraspProject provides a simple model of the hand and arm, with three degrees of freedom in the arm

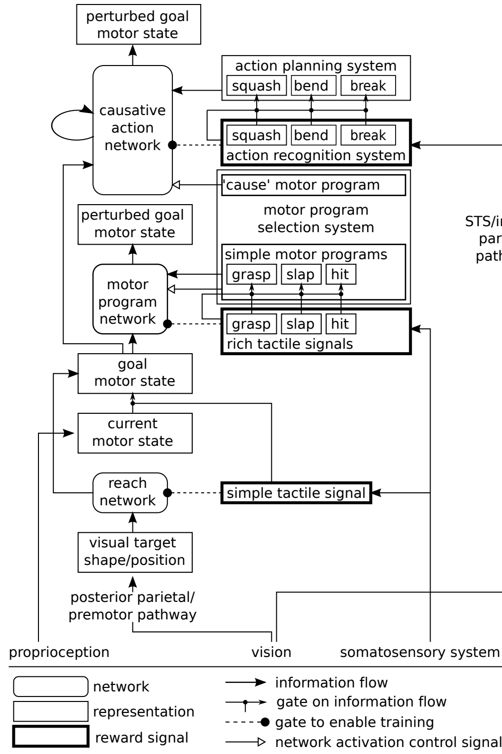


Figure 3: Architecture of the motor control network

(two at the shoulder and one at the elbow) and one in the hand controlling grip aperture (see Figure 2a). It also provides a fairly rich model of the touch sensors in the fingers. Finger pads are modelled as deformable grids of rigid bodies connected by springs. (A single finger pad in light contact with a solid surface is shown in Figure 2b.) Information about light touches is provided by collision detectors on each pad, and information about stronger touches which deform the surface of the skin is read from the joint angles between adjacent pads.

Architecture of the motor control network

Our model of the motor system is a neural network for learning hand actions directed at target objects. It provides a simple model of some aspects of infant motor development.

The general architecture of the network is shown in Figure 3. It consists of three sub-networks arranged in sequence. These are assumed to be trained at three successive developmental stages, by reward signals of different degrees of complexity. In this scheme, the system is initially rewarded by very simple sensory signals, which train a simple motor circuit, but as learning takes place in this circuit, more complex reward signals become available, which in turn train higher-order circuits. The first two networks are described in detail in Lee-Hand *et al.* (2012), and their interaction with the third network is described in Lee-Hand (2013).

The first network to be trained is called the **reach network** (see the lower part of Figure 3). This network learns a function which maps a visual representation of a target object onto a goal motor state of the hand and arm. (The visual representation has two components, one relating to the position of the target, the other to its shape. The former representation maps to a goal arm state; the latter to a goal hand state.) During training, the agent visually attends to objects in its perispace, and executes hand/arm actions at random. Sometimes these actions result in its hand touching the target, evoking a touch signal (the simple touch signal). This signal is intrinsically rewarding (as in Oztop *et al.*, 2004). The touch signal has two functions. First, it allows a proprioceptive representation of the agent's current motor state to be copied into the medium holding its goal motor state (see the upper arrow leading from the simple tactile signal). Second, it allows the reach network to be trained, so that the current visual representation of the target object is associated with this newly specified goal motor state, and similar presentations of the target in the future will automatically elicit an appropriate motor goal.

This simple circuit implements a particular version of Hommel *et al.*'s model of event codes. Learning in the circuit creates what can be thought of as a single simple action category, associated with the sensory representation of a touch to the hand: a network which maps visual stimuli onto motor goals which will bring about this representation. Motor goals in the circuit are associated with sensory stimuli in three ways. Any representation in the motor goal medium is implicitly associated with one particular reward stimulus (a simple touch sensation). Specific motor goals are associated axiomatically with specific motor states (sensed proprioceptively) when the reward stimulus is evoked. And specific motor goals are also associated through learning with arbitrary sensory stimuli (in this case visual), which carry information about the motor states associated with reward signals. Again this happens at the time the reward stimulus is evoked. The key devices in the circuit are reward-gated learning and copy operations. These devices will be replicated in the other two networks.

The reach network generates a motor goal—but of course there must also be a mechanism which achieves this goal. At the first developmental stage, we assume this mechanism is a simple feedback motor controller. This device takes the current motor state and the goal motor state and generates a motor signal proportional to the difference between them, in a direction which reduces this difference. (The controller is not shown in Figure 3.) A feedback controller does not need to be trained; it can be assumed to be present at birth. (We use a PID controller; see e.g. Araki, 2006). However, mature motor control involves a mixture of feedback control and *feedforward* control (see e.g. Kawato *et al.*, 1987). Feedforward control exploits learning about the properties of the agent's motor system to optimise action trajectories. If we think of the feedforward controller in sufficiently general terms, we can say that it is through learning in this controller that an agent can acquire a repertoire of different action categories. Different actions (like grabbing or punching or slapping) have different characteristic trajectories of the hand and fingers; the feedforward control system somehow learns about the distinct effects of particular trajectories and creates action categories associated with each. However, it is not clear how different trajectories are represented in the biological motor control system. There is good evidence that agents do not compute detailed trajectories in advance; these are only generated 'on the fly', as an action is actually underway (see e.g. Cisek, 2005). Our network implements a particular idea about how trajectories are represented. We assume that the agent evoking a goal motor state can generate learned *perturbations* of this goal state as an action is under way, which deviate the hand from the normal course it would take under simple feedback control. For instance, to generate a trajectory bringing the hand onto the target from above, the goal state could be temporarily perturbed to a point above the target, so the hand initially moves higher than it would normally do. This idea is discussed in more detail and evaluated in Lee-Hand *et al.* (2012). This kind of learning takes place in the second network in our model, the **motor program network** (see the middle of Figure 3).

The motor program network learns to map a goal motor state onto a perturbed goal motor state, which is applied at the start of a reach action and removed when the hand is at a specified distance from the target. Learning in this network begins when the reach network reliably generates actions that lead to reward signals. During training, random perturbations are applied to the goal motor state produced by the reach network. From time to time, these perturbations result in richer tactile reward signals than those used to train the reach network. There are several different signals, which result from particular perturbations. Some perturbations result in a grasp or near-grasp, which generates a characteristic rich tactile stimulus. Others result in a slap movement, which generates another, different, tactile stimulus. (These rich stimuli are almost never generated through pure feedback control, because they result from special trajectories.) When a rich tactile stimulus is generated, copy and learning operations take place in the motor program circuit which are analogous to those in the reach circuit. First, the tactile stimulus is copied to an area holding 'motor programs'. Second, the motor program network is trained to map the current goal motor state, *plus the currently active motor program*, onto the perturbation which resulted in the reward. After this learning, activating a specific motor program will generate an action with a characteristic trajectory. We envisage motor programmes competing with one another, with the winner being selected.

Note that the motor program network must execute in parallel with the simple reach network. It basically modulates the behaviour of the simple network, in a manner reminiscent of Brooks' (1991) subsumption architecture. In order to execute a motor program, it is important that the whole motor program circuit is enabled, or turned on. Accordingly, while different motor programs provide different input to the motor program network, they also uniformly generate a control signal to enable the network they provide input to.

The final network to be trained is the **causative action network** (see the top of Figure 3). Our assumption here is that there is a higher level of motor control where sensory reward signals are generated within a perceptual module whose primary function is to classify actions observed occurring in the external world. There is a well-studied perceptual module of this kind in the brain, implemented in a pathway from sensory cortices (in particular visual cortex) through the superior temporal sulcus (STS) and inferior parietal cortex to the premotor cortex (see e.g. Keysers and Perrett, 2004). When an agent allocates attention to an external object, representations in this pathway encode the actions of this object in various ways. Canonically, the action recognition pathway is active when an agent is passively observing the external world. But consider what happens when the agent is attending to an external object *as a target*, while directing a hand action towards it. Any actions evoked in the action recognition pathway in this scenario are potentially actions *brought about by the hand action*. We propose that during action execution, action signals evoked in the action recognition pathway *function as reward signals*, which train the causative action network to bring about particular distal actions in the world.

Training in this higher-level motor circuit again proceeds by random generation of perturbations to the goal motor state delivered by the reach network. In this circuit, *sequences* of perturbations are applied, to generate still more complex trajectories. (This is depicted in the diagram by a recurrent input, though in our implementation we 'unroll' this recurrence and generate exactly two perturbations.) Some of these sequences cause particular patterns of movement in the target object, which are interpreted as external actions by the action recognition system. Activation of an action representation in the action recognition system when performing an action on a target object is hard-wired to generate a reward signal. This signal has two effects. First, the observed action is copied to a specialised motor medium: specifically, a medium in which action plans are held. Second, the causative action network is trained to map the basic goal motor state delivered by the reach network onto the sequence of perturbations which led to reward. Note that the network also takes representations in the action planning system as input. After training, the causative action network can take a simple goal motor state, plus an action representation in the action planning system, and generate a sequence of perturbations which will lead to observation of a specific action on the attended target. And different patterns in the action planning system will lead to different observed actions.

This network enables a rich repertoire of actions to be learned. It preserves Hommel *et al.*'s idea that action representations are organised around their perceptual effects. But since the action recognition network generates rich, high-level percptual signals, a correspondingly rich set of motor programs can be established. At the same time, the basic mechanisms through which learning happens are the same as in much simpler motor learning systems.

Part of the design of the causative action circuit is that 'cause' is motor program in its own right, which competes within the motor program selection system against regular motor programs like 'grasp' and 'slap'. One important difference is that the 'cause' action enables the causative actions network rather than the motor program network, but other than that it counts as a regular motor program. This raises some important questions about how causative actions are planned and executed. When an agent decides to perform a causative action, presumably he has some particular caused action in mind. But at the time of planning, this caused action is in the future: minimally, the agent must bring his hand into contact with the target object before he can cause it to move in any way. Moreover, there is hardly ever a clear way of decomposing a causative action into a simple reach action and a subsequent manipulation. In order to cause a particular action in a target object, the trajectory of the hand towards the object must typically be biased from the very start: for instance, to cause an object to squash, the hand must approach the target from a particular direction, and with particular force. So the movements which bring about the caused action must be initiated some time before the action is perceived.

Our way of addressing this issue in the network is to activate the motor correlates of perceived actions in a medium holding *planned actions*, rather than in the medium of regular motor programs like 'grasp' and 'slap'. An underlying assumption in our model is that an agent brings about actions through planned sequences of sensory or motor operations (for details see Knott, 2012). We also assume that planned sequences are selected as wholes, and that the component actions in a planned action sequence are active in parallel in the working memory medium where actions are planned. (This assumption is well supported by single-cell recordings in monkeys; see e.g. Averbeck *et al.*, 2002). When the causative actions network is exploring causative actions, it will activate the 'cause' motor programme experimentally, and choose a random sequence of perturbations. In some cases, this results *some time later* in activation of an action in the action recognition system: say 'squash'. This observed action activates a corresponding planned action; additionally the sequence-planning system will learn that the sequence 'cause', 'squash' is a good one to execute in the current context, so that when a similar context occurs in the future, it will activate this planned sequence. Now consider what happens when the planned sequence is executed. The agent first executes the motor programme 'cause'. This enables the causative action network, which generates a sequence of perturbations. Crucially, the causative action network also takes input from the planning medium in which the caused action ('squash') is active as part of the planned sequence. So as soon as it is initiated, the network is configured to generate the perturbation sequence which led to the caused action, even before this action actually occurs.

The key mechanism enabling causative actions to be executed is one which activates a sensory representation (the squash action) *as a goal* some time before it is evoked as a sensory stimulus. Note that something very similar happens in the other networks; for instance in the reach network the actual motor state where the touch sensation occurs is activated as a goal motor state. In the simple network this activation is possible because visual perception provides information about reward-associated motor states. In the higher-level causative actions network, the advance notification of reward comes from the working memory system which stores prepared actions. But the effect is much the same.

Experiments learning causative actions

We built two objects in the simulation environment which could undergo specialised kinds of action. One comprised two horizontal planes connected by a spring; this object could be 'squashed' by pushing down on it. The other was a lever which could pivot around a joint; this object could be bent. Our action recognition system consulted the states defined in the physics engine directly; we did not attempt to simulate the action recognition module (though we did simulate the visual inputs to the reach network in a simple way). Each training trial involved the presentation of a single object (either the bendable target, the squashable target or a rigid object) in one of several possible positions. This led to activity in the nework and a motor action. If the rigid object was presented, 'grasp' was activated in the motor program circuit, so that this circuit could be trained; otherwise the activated program was always 'cause', the operation activating the causative action network. This network always generated two perturbations in sequence. At the start of training, perturbations were annealed with noise; this was gradually reduced as the network learned. If the action recognition system detected an action A, it generated a reward, and the sensed action activated a corresponding unit in the action planning system, resulting in the sequence ['cause', A]. This sequence plan was also associated with the visual target shape, so subsequent presentation of this target would activate the plan. As training progressed, the system learned perturbations which brought about particular perceived actions, and also learned to map visual target representations onto appropriate sequence plans. Details of the network's training and testing are given in Lee-Hand (2013).

A syntactic analysis of causative actions, and a sensorimotor interpretation

It is interesting to compare the structures used by our network to learn and generate causative actions with the syntactic structure of sentences reporting causative actions. As discussed at the start, a Chomskyan analysis posits an underlying level of syntactic representation at which the sentence *John bent the lever* contains an explicit 'cause' action as its main verb, which takes as its argument a nested action ('The lever bent'). The network model also contains an explicit 'cause' operation (the 'cause' motor programme), and a nested action (the action delivered by the action recognition system). An emerging school of thought in cognitive science is that concrete sentences get their meanings by evoking embodied representations in the sensorimotor system (see e.g. Glenberg and Kaschak, 2002; Barsalou, 2008). The network model of causative actions presented above may support an interesting 'embodied' account of the semantics of causative transitives like *John bent the lever*. We will conclude by discussing this possibility.

The underlying structure of sentences in Chomsky's (1995) Minimalist model is called 'logical form' (LF). The LF of *John bent the lever* is shown in Figure 4. LF representations have a right-branching recursive structure: the units of recursion, called X-bar schemas (XPs), are indicated with boxes in the figure. (Details of the higher two XPs are omitted for simplicity.) Knott (2012) uses Minimalist LF structures to express a strong claim about the embodied nature of sentence meanings. In his proposal, the LF of any sentence reporting a concrete episode in the world can be interpreted as a description of the sensorimotor processes through which this episode is experienced. Knott assumes Ballard *et al.*'s (1997) account of sensorimotor processing, which posits that this processing is organised into well-defined sequences of attentional or motor operations called 'deictic operations'. The key idea in Knott's proposal is that the LF structure of a concrete sentence describes a sequence of deictic operations—specifically, that each XP in the structure describes a single operation. The sensorimotor denotations of XPs are shown in red in Figure 4.

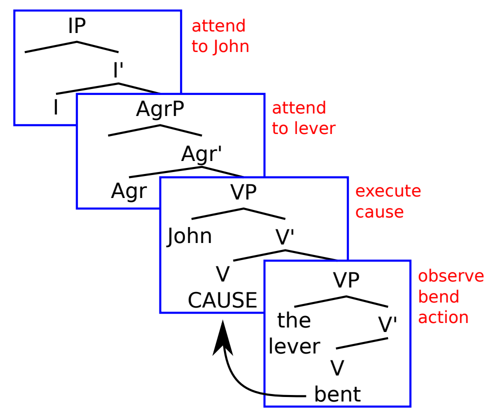


Figure 4: LF of *John bent the lever*, including head-raising operation

This general proposal makes a specific prediction about the sequential structure of a causative action. As shown in Figure 4, the XP introducing CAUSE immediately dominates the XP presenting the nested action. Knott's proposal thus predicts that a causative action involves two stages: activation of a 'cause' operation, followed by experience of the 'bend' action. And indeed, execution of causative actions in our network model has this sequential character. So the recursive structure of LF has the right general form.

However, an additional neat correspondence can be drawn between the structure of LF and the structure of the sensorimotor routine. As discussed above, the hand movement initiated by the 'cause' motor program must follow different trajectories to the target to achieve different causal effects on it. In the network model we catered for this by having the causative actions network take input from a medium representing a casual effect *as it is planned*, rather than as it is later observed. We assumed that this medium holds the entire sequence of actions being executed by the agent, as a sustained signal, so as soon as the causative network is engaged, its output is influenced by the planned action to be brought about. Now note that in the LF structure in Figure 4, the lower verb *bent* raises to the position of the higher verb CAUSE (as shown by the arrow) so that it can be pronounced in this higher position, giving the surface form *John bent the cup*. We propose to explain the Minimalist device of verb raising in sensorimotor terms by assuming in general that surface verbs describe motor actions *as they are planned* rather than as they actually occur. The reason why the verb *bent* can be pronounced at the higher verb position is that it denotes a signal in the planning medium, which is tonically active through the whole executed routine. In fact, this account of verb raising follows naturally from a wider sensorimotor account of verb raising which was proposed by Knott (2012) and is implemented in a neural network model of sentence generation described in Takac *et al.* (2012). But in the current context, the key point is that the sensorimotor interpretation of LF explains both the structural relationship between the CAUSE verb and its complement VP and the extended syntactic domain of the nested verb *bend* at LF which allows it to appear in the position of the CAUSE verb in surface structure.

Summary

In this paper we presented a neural network model of the learning and execution of causative actions. The model embodies a particular take on Hommel *et al.*'s proposal that actions are defined in terms of their effects: a basic circuit implementing this principle is replicated in three different components of the network, at different levels of abstraction. At the same time, the network provides the basis for an interesting account of the syntactic structure of causative sentences—specifically, of the relation between a cause predicate and the action which this predicate introduces.

References

Araki, M. (2006). PID control. In H. Unbehauen, editor, *Control Systems, Robotics and Automation Vol. II*.

Averbeck, B., Chafee, M., Crowe, D., and Georgopoulos, A. (2002). Parallel processing of serial movements in prefrontal cortex. *PNAS*, **99**(20), 13172–13177.

Ballard, D., Hayhoe, M., Pook, P., and Rao, R. (1997). Deictic codes for the embodiment of cognition. *Behavioral and Brain Sciences*, **20**(4), 723–767.

Barsalou, L. (2008). Grounded cognition. *Annual Review of Psychology*, **59**, 617–645.

Brooks, R. (1991). Intelligence without representation. *Artificial Intelligence*, **47**, 139–159.

Chomsky, N. (1995). *The Minimalist Program*. MIT Press, Cambridge, MA.

Cisek, P. and Kalaska, J. (2005). Neural correlates of reaching decisions in dorsal premotor cortex: Specification of multiple direction choices and final selection of action. *Neuron*, **45**, 801–814.

Glenberg, A. and Kaschak, P. (2002). Grounding language in action. *Psychonomic Bulletin and Review*, **9**(3), 558–565.

Hommel, B. (1993). Inverting the Simon effect by intention. *Psychological Research*, **55**, 270–279.

Hommel, B., Müsseler, J., Aschersleben, G., and Prinz, W. (2001). The theory of event coding (TEC): A framework for perception and action learning. *Behavioral and Brain Sciences*, **24**, 849–878.

Kawato, M., Furawaka, K., and Suzuki, R. (1987). A hierarchical neural network model for the control and learning of voluntary movements. *Biological Cybernetics*, **56**, 1–17.

Keysers, C. and Perrett, D. (2004). Demystifying social cognition: A Hebbian perspective. *Trends in Cognitive Sciences*, **8**(11), 501–507.

Knott, A. (2012). *Sensorimotor Cognition and Natural Language Syntax*. MIT Press, Cambridge, MA.

Lee-Hand, J. (2013). A neural network model of causative actions. MSc thesis, Dept of Computer Science, University of Otago.

Lee-Hand, J., Neumegen, T., and Knott, A. (2012). Representing reach-to-grasp trajectories using perturbed goal motor states. In *Proceedings of the Pacific Rim Conference on Artificial Intelligence (PRICAI)*, pages 250–261.

Matsumoto, K., Suzuki, W., and Tanaka, K. (2003). Neuronal correlates of goal-based motor selection in the prefrontal cortex. *Science*, **301**, 229–232.

Neumegen, T. (2013). A computational platform for simulating reach-to-grasp actions: modelling physics, touch receptors and motor control mechanisms. MSc thesis, Dept of Computer Science, University of Otago.

Oztop, E., Bradley, N., and Arbib, M. (2004). Infant grasp learning: a computational model. *Experimental Brain Research*, **158**, 480–503.

Prinz, W. (1997). Perception and action planning. *European Journal of Cognitive Psychology*, **9**(2), 129–154.

Simon, J. (1969). Reactions towards the source of stimulation. *Journal of Experimental Psychology*, **81**, 174–176.

Takac, M., Benuskova, L., and Knott, A. (2012). Mapping sensorimotor sequences to word sequences: A connectionist model of language acquisition and sentence generation. *Cognition*, **125**, 288–308.

Umiltà, M., Escola, L., Intskirveli, I., Grammont, F., Rochat, M., Caruana, F., Jezzini, A., Gallese, V., and Rizzolatti, G. (2008). When pliers become fingers in the monkey motor system. *PNAS*, **105**(6), 2209–2213.