On the dynamic interplay between perception and action - a connectionist perspective

Haazebroek, P.

Author: Haazebroek, Pascal
Title: On the dynamic interplay between perception and action : a connectionist perspective
Issue Date: 2013-12-11

Chapter 2
HiTEC Connectionist Model

This chapter is an integration of major parts of the following articles:

Haazebroek, P., Raffone, A., & Hommel, B. *HiTEC: A Connectionist Model of the Interaction between Perception and Action Planning.* Manuscript submitted for publication.

Haazebroek, P., van Dantzig, S., & Hommel, B. (2013). How task goals mediate the interplay between perception and action. F*rontiers in Psychology, 4:247*.

In this chapter we describe the HiTEC connectionist model in full detail. We start out with discussing the general cortical layering of the brain and our general connectionist modeling approach. Then we describe the specific HiTEC architecture, followed by its computational implementation. We proceed with discussing how HiTEC allows simulating behavioral studies and, finally, we compare our approach with related work and address how neuron-like representations may realize stimulus-response translation.

Cortical layering

Neurons in the primate cortex appear to be organized in numerous interconnected cortical layers. It is commonly assumed that this organization allows the brain to encode perceived objects in a *distributed* fashion. That is, different features seem to be processed and represented across different cortical layers (e.g., Cowey, 1985; DeYoe & Van Essen, 1988), coding for different perceptual modalities (e.g., visual, auditory, tactile, proprioceptive) and different dimensions within each modality (e.g., visual color and shape, auditory location and pitch). Each sensory cortical layer typically contains neurons that are responsive to specific sensory features (e.g., a specific color or a specific visual location). Cortical layers in the motor cortex contain neurons that code for more or less specific movements (e.g., the muscle contractions that produce the movement of the hand pressing a certain key, or more complex movement such as shifting one's weight to the right). Higher up in the processing stream there are cortical layers containing neurons that are receptive to stimulation from different modalities. In effect, they are considered to integrate information from different senses and modalities. Finally, the prefrontal cortex contains neurons that are involved in task-generic cognitive control (Duncan & Owen, 2000). These *levels of representation* are illustrated in Figure 3 and form the basis of the HiTEC model architecture.

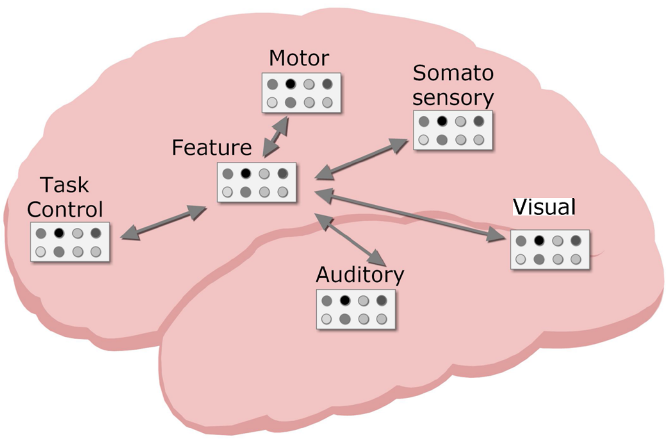

Figure 3. Tentative locations of various cortical layers in the primate brain with sensory layers in sensory regions, task control layers in the frontal lobe, motor layers in motor area and intermediate feature layers mediating between lower and higher region layers.

Crucially, cortical layers are not only interconnected by feedforward connections (i.e., from lower to higher level layers) but there are also dense neural pathways from centers of higher brain function back into perception centers (Braitenberg & Schüz, 1991; Young, 1995) suggesting *top-down influence* of higher level layers on processing within lower level layers (e.g., Prinz, 2006). This constraint of reciprocal connectivity between various levels of representation is taken seriously in the HiTEC connectionist model.

Connectionist approach

The cortical layers in the primate brain contain a vast amount of spiking neuron cells. The local interactions between these neurons are largely random, but on a group level – a neuron population – the global population activity (i.e., mean spike frequency) can be considered deterministic (Wilson & Cowan, 1972). That is, mean activation depends on various inputs and the decay of the neuron population (see Figure 4).

To model such neuron populations HiTEC follows[2] the interactive activation connectionist modeling approach (PDP; Rumelhart et al., 1986). In these connectionist models processing occurs through the interactions of a large number of interconnected elements called units. In HiTEC, these units may stand for the neuron populations described above and are organized into higher structures representing cortical layers. Each unit has an activation value indicating local activity. Processing occurs by propagating activity through the network; that is, by propagating activation from one unit to the other, via weighted connections. When a connection between two units is positively weighted, the connection is excitatory and the units will increase each other's activation. When the connection is negatively weighted, it is inhibitory and the units will reduce each other's activation. Processing starts when one or more units receive some sort of external input. Gradually, unit activations rise and propagate through the network while interactions between units control the flow of processing. Some units can be designated output units. When activations of these units reach a certain threshold the network is considered to produce the corresponding output(s).

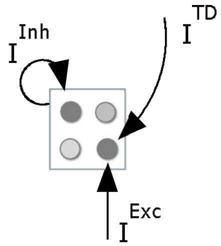

Figure 4. Cortical layer with neuron populations with various inputs (TD: top down, Inh: lateral inhibition, Exc: excitatory input)

[2] For our current work we have focused on representations and interactive processing. For this purpose, PDP principles provided the means we needed. This modeling framework, however, is not essential for TEC/HiTEC. Other aspects are probably hard to tackle within the limitations of PDP, such as binding and integration. In the future we may change to or add other modeling principles than PDP prescribes.

HiTEC architecture

HiTEC has a multiple-layer architecture (see Figure 5) and recurrent interactions at multiple levels, including feedback to lower level units. In HiTEC feedforward and feedback interactions are cooperative and lateral interactions (i.e., within layers) are competitive (see also Murre, Phaf & Wolters, 1992; van Dantzig, Raffone & Hommel, 2011). The HiTEC neural network is composed of excitatory and inhibitory neural units in each layer. The coding functions are implemented as excitatory units[3]. The inhibitory units are only involved in lateral competitive interactions; by contrast, the excitatory units can receive inputs from and send outputs to associated units in other layers, yielding cooperative interactions. Within each layer inhibitory units are activated by an associated excitatory unit and propagate inhibition to the excitatory units that implement other codes in the same layer (see Figure 6).

We now first describe the general model architecture, and then describe the model behavior and the computational specification of the network units. HiTECs general architecture contains sensory layers, feature layers, a task layer and a motor layer, as depicted in Figure 5. Each layer resembles a cortical circuitry and contains codes implemented as excitatory connectionist network units as described above. The different codes (and related units) are characterized as follows.

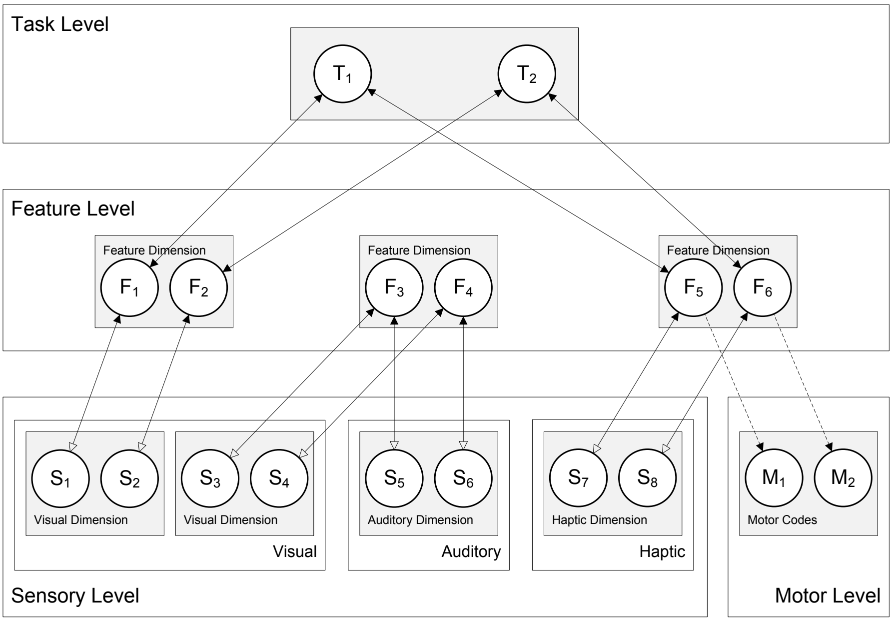

Figure 5. General computational structure of HiTEC. Codes are contained in layers at various levels, and are connected by excitatory connections. Solid lines denote fixed weights, dashed lines are connections with learned weights. Sensory codes receive modulated excitatory input from feature codes, denoted by the open arrows. Note that feature code – motor code associations are one-way connections and that feature code – task code connections are non-modulated both ways.

[3] We have opted for localist representations to keep HiTECs architecture and representations as simple as possible. There is, however, nothing that precludes the possibility that any of the codes could be distributed over many sub-codes/sub-units.

Sensory codes

In HiTEC, different perceptual modalities (e.g., visual, auditory, tactile, proprioceptive) are distinguished and different dimensions within each modality (e.g., visual color and shape, auditory location and pitch) are processed and represented in different sensory layers. Each sensory layer contains a number of sensory codes that are responsive to specific sensory features (e.g., a specific color or a specific location in the visual field). Sensory codes receive external input and feedback activation from feature codes.

Crucially, the responsiveness of sensory coding units is modulated by connected feature coding units. This is realized by making the inputs from feature units to a sensory coding unit dependent on that sensory coding unit's activation, which is primarily determined by its external stimulation. This way, a sensory coding unit cannot become highly active by mere top down input, which would be the equivalent of a hallucination.

Motor codes

The motor layer contains motor codes, referring to more or less specific movements (e.g., the movement of the hand pressing a certain key or producing a verbal utterance). Although motor codes could also be organized in multiple layers (e.g. reflecting different body parts), in the present version of HiTEC we consider only a single basic motor layer with a set of motor codes. Motor codes are activated by feature codes. When the activation level of one of the motor coding units reaches a set response threshold, the motor code is assumed to be selected and executed. Subsequent *action effects in the environment* are presented to the sensory coding units.

How motor actions are controlled is more explicitly addressed in Chapter 3. Note that our present account of motor information represents a dramatic simplification. Movements are unlikely to be represented by coherent, encapsulated motor programs (as considered by Keele, 1968) but, rather, in a rather complex, distributed fashion (Hommel & Elsner, 2009; Wickens, Hyland, & Anson, 1994). However, this simplification does not affect our main arguments and it helps keeping the model and its behavior reasonably transparent.

Feature codes

TEC's notion of feature codes (Hommel et al., 2001) is captured at the feature level by codes that are connected to and thus grounded in both sensory codes and motor codes. Crucially, the same (distal) feature code (e.g., 'left') can be connected to multiple sensory codes (e.g., 'left proprioceptive direction' and 'left visual shape'). Thus, information from different sensory modalities and dimensions is combined in one feature code representation. It is assumed that feature codes arise from regularities in sensorimotor experience, presumably by detecting co-occurrences of sensory features. The distal feature 'left', for example, could arise from perceptual experience of numerous objects that were visible and audible on the left. Future encounters of objects audible on the left activate the 'left' feature code which – by means of its connections to both 'left auditory location' and 'left visual location' – will enhance the processing of visual left locations. In other words, hearing something on the left will result in expecting to see something on the left as well, which seems to be quite useful, for example when visual sensory input is degraded. Although feature codes are considered to arise from experience, in the present HiTEC version we assume the existence of a set of feature codes (and their connections to sensory codes) to bootstrap the process of extracting sensorimotor regularities in interactions with the environment.

Since feature codes connect to both sensory codes and motor codes, they can be considered common codes in the sense of Prinz (1990), subserving both stimulus perception and response planning. When a certain feature code is used to represent a task stimulus and this same feature code is also used to represent a task response, the resulting code overlap may result in compatibility effects. Such compatibility effects are demonstrated in the simulations discussed in the next chapters, most notably in Chapter 4.

Task codes

The task layer contains generic task codes that reflect alternative stimulus-response combinations resulting from the task context. Different task codes reflect different stimulus-response choice options within the task context. Task codes connect bi-directionally to feature codes, both the feature codes that represent stimuli and the feature codes that represent responses, in correspondence with the current task context. Note that task codes themselves are task-generic (i.e., labeled 'T_1', 'T_2' et cetera); their meaning derives from their connections with specific feature codes.

The multiple-layer recurrent neural network architecture with different types of codes and the connections between associated codes is illustrated in Figure 5. Note that the connection weights can be different (asymmetrical) for corresponding 'forward' and 'backward' connections (e.g. different weights for the connection from feature codes to task codes, and the reciprocal connection from task codes to feature codes).

Basic model behavior

The presentation of a stimulus is simulated by feeding external input to the appropriate (excitatory) sensory codes. This results in a gradual increase of their activation level, which is translated into output to feature codes. Thus, activation flows gradually from sensory codes to (stimulus related) feature codes to task codes to (response related) feature codes to motor codes. Once a motor code is activated strongly enough it is assumed to lead to the execution of a motor response to the presented stimulus. The gradual passing of activation between codes in different layers along their connections is iterated for a number of simulation cycles, which allows for the simulation of reaction time (i.e., number of processing cycles from stimulus onset to response selection). Crucially, activation also propagates back from task codes to stimulus related feature codes that in turn modulate the sensitivity of sensory codes, thereby rendering an integrated processing system with both feedforward and feedback dynamics rather than a serial stage-like processing mechanism.

Ideomotor learning

In HiTEC, connections between feature codes and motor codes are learned according to the ideomotor principle (Hommel, 2009; James, 1890; Lotze, 1852). This principle states that when one executes a particular action and perceives the resulting effects in the environment, the active motor pattern is automatically associated to the perceptual input representing the action's effect. Based on these action-effect associations, people can subsequently plan and control a motor action by anticipating its perceptual effect.

In similar vein, learning in HiTEC is done by first randomly activating motor codes, not unlike the random movement behavior of newborn infants (motor babbling) or complete novices at a new task. When a motor code reaches a threshold of activation, we assume that the response is executed, resulting in perceivable changes in the environment (action effects). Perceiving these action effects constitutes stimulating the respective sensory codes; activation is subsequently propagated from these sensory codes towards feature codes (cf. Elsner & Hommel, 2001). Finally, associations are learned between these feature codes and the executed motor code. During subsequent stimulus-response translation these associations enable activation of the appropriate motor action by activating the associated feature codes. Thus, a motor action can be selected by 'anticipating its perceptual effects'. Ideomotor learning and its role in action control is addressed more elaborately in Chapter 3.

Task internalization

In behavioral experiments both stimuli and responses can have a variety of features. The task context dictates which of these features are relevant (i.e., the features to look for and to discriminate) and which are irrelevant. In HiTEC, a task instruction is implemented by connecting feature codes and task codes according to the actual task rules in terms of stimulus features and response (i.e., action effect) features. This procedure allows the task instruction to be readily internalized. An example task instruction "when you hear a high tone, press the left key" would then be implemented as connections from 'High' to 'T_1' and from 'T_1' to 'Left' and 'Key'. During the subsequent stimulus-response translation, these connections modulate the responsiveness of feature codes to bottom-up input from stimulated sensory codes and through these connections activation is propagated towards feature codes associated to the proper motor responses in accordance with task demands (cf., Miller & Cohen, 2001). This way, appropriate goal oriented behavior can take place within a certain task context.

In the present HiTEC version, these connections between feature codes and task codes units are set by hand in correspondence with the verbal task instruction. However, it is conceivable that these connections arise from external or internal verbal or nonverbal (self-) instruction and are maintained due to internal motivational drives. We hypothesize that feature codes could be accessed by means of verbal labels and that receiving a task instruction would activate these feature codes (e.g., Bargh & Gollwitzer, 1994; Hommel & Elsner, 2009; Logan & Bundesen, 2004) and connect them to generic task codes (i.e., some sort of internal simulation of the translation from stimulus features to response features). Note that apart from this instruction based wiring we do not assume any other type of task-specific addition to the model. That is no additional 'task inputs' or biases in code dynamics are required to control stimulus-response translation.

Computational implementation

HiTEC codes are implemented as (excitatory) neural network units, characterized by an activation level. These units, which may stand for neuronal groups, receive excitatory and inhibitory inputs from other units and background noise. Excitatory inputs can either be voltage independent or voltage dependent, i.e. with a modulatory role dependent on the voltage ('activation') of the receiving unit. Indeed, cortical feedback connections are generally voltage dependent, i.e. necessitate a sufficient level of feedforward (stimulus related) synaptic input to be effective. In addition, the activation of the units is characterized by a decay rate, so that in case of absence of any input the activation will decay exponentially towards a resting level. Units in the sensory layers can also receive an external (stimulus related) input. Thus, on every cycle unit activations are updated according to the following equation:

$$A_i(t+1) = (1-d_a) \times A_i(t) + \gamma_{exc} \times Exc_i \times (1 - A_i(t)) + \gamma_{inh} \times Inh_i \times A_i(t) \qquad (1)$$

In this equation, d_a is the activation decay rate, $A_i(t)$ is the activation level of unit i at time t, Exc_i is the sum of its excitatory input, Inh_i is its inhibitory input and both γ_{exc} and γ_{inh} are scaling terms. Note that both excitatory and inhibitory inputs are scaled in a way that the unit's activation may take on any real value between 0.0 and 1.0. The excitatory input is computed as follows:

$$Exc_i = ExcVI_i + ExcVD_i + Ext_i + Noise_i \qquad (2)$$

Here, $ExcVI_i$ is a voltage independent ('non-modulatory') input from other units in the network, which does not depend on the activation of the receiving unit; $ExcVD_i$ is a voltage dependent input, which is instead dependent on the activation of the receiving units (implicitly related to the membrane potential of receiving neurons). These different excitatory inputs stand for different synaptic currents in cortical networks: feedforward signaling takes place by voltage-independent synaptic currents, and feedback signaling by modulatory voltage dependent currents (e.g., Dehaene et al., 2003; Raffone & Pantani, 2010; Tononi, Sporns, & Edelman, 1992). Ext_i is input from external stimulation (only for units in the sensory layers) and $Noise_i$ is a noise term. This noise term is determined by drawing a random value from a Gaussian distribution[4] at each update cycle and for each unit independently.

[4] Determining the noise term by drawing from a Gaussian distribution sometimes (with our parameters, in < 5% of the cases) results in a negative value. In order to restrict the excitatory input to positive values, we replace any negative value by 0.0.

The voltage independent input is obtained by calculating the weighted sum of the outputs of all connected units (apart from units where voltage dependent input applies, see below):

$$ExcVI_i = \varphi \sum_k w_k^+ \; F(A_k(t) \tag{3}$$

Here, w^+ are the positive weights of the connections from other units k to unit i and φ is a scale factor. The output of a unit is a non-linear function of its activation value, using the following function (Grossberg & Grunewald, 1997; Grossberg & Somers, 1991), with parameters *na* and *qa*:

$$F(A_i) = \frac{A_i^{na}}{(qa)^{na} + A_i^{na}} \tag{4}$$

Crucially, the responsiveness of sensory coding units is modulated by connected feature coding units. This is realized by making the inputs from feature units to a sensory coding unit dependent on the sensory coding unit's activation, which is primarily determined by its external stimulation. This way, a sensory coding unit cannot become highly active by mere top down input. This voltage dependent input from feature coding units to sensory coding units is computed using the following equation (see Tononi et al., 1992, for a similar computation):

$$ExcVD_i = \sum_k w_k^+ \; F(A_k(t)) \times \frac{\max(A_i(t) \times (1 - d_a) - VT, 0)}{1 - VT} \tag{5}$$

Here, d_a is the activation decay rate and VT is the voltage threshold. When the sensory coding unit has a (scaled) activation level higher than this threshold, top down input from connected feature coding units is taken into account, rescaled in proportion to the voltage threshold and added to the sensory coding unit's excitatory input. If the sensory coding unit's scaled activation level is lower than the voltage threshold, this input is discarded.

Activation of units is competitive, so that coding units within the same layer (sensory layers, feature layers, task layer, or motor layer) inhibit each other. This is computationally realized by the involvement of 'paired units'. As shown in Figure 6, each of the inhibitory units receive activation from its excitatory paired unit, and propagates inhibition (i.e., their 'outgoing' connections are negatively weighted) to all other excitatory units within the same layer. Such inhibition is characterized by non-linearity, i.e. inhibitory units propagate inhibition when they approach a level of activation. This mechanism ensures that within a layer only one unit becomes highly active after a certain number of simulation cycles.

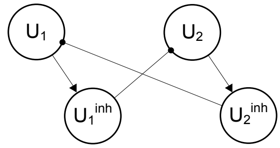

Figure 6. Inhibition between units within the same layer. In each layer, codes are implemented as excitatory units with additional paired inhibitory units. These inhibitory units receive activation from their excitatory paired unit (arrowed connections) and send inhibition (i.e., activation through negatively weighted connections; denoted with solid discs) to all other excitatory units within the same layer.

Inh_i is computed using the following equation:

$$Inh_i = \sum_k w_k^- \; F(A_k(t)) \tag{6}$$

Here, k denotes the inhibitory units belonging to any other unit than unit i in the layer, and w^- are the negative connection weights. The activation of inhibitory units is updated in a similar fashion as the excitatory units, but their input can only be excitatory from the associated paired unit.

Connections

Weights between sensory coding units and feature coding units are set by hand as are the weights of the connections between feature coding units and task coding units closely following the task instruction. The weights from feature coding units to motor coding units are modified using Hebbian learning. Specifically, at the end of each learning trial (see below), the connection weights from feature coding units to motor coding units are updated during a number of cycles according to the following set of equations:

$$\begin{aligned} w_{jk}(t+1) &= (1-d_w)\times w_{jk}(t) + LR \times Act_j(t) \times Act_k(t) \times \left(1-w_{jk}(t)\right) \\ Act_j(t) &= \frac{A_j(t)-LT}{1-LT} \quad if\ A_j(t) > LT \\ Act_j(t) &= 0 \quad if\ A_j(t) \le LT \\ Act_k(t) &= \frac{A_k(t)-LT}{1-LT} \quad if\ A_k(t) > LT \\ Act_k(t) &= 0 \quad if\ A_k(t) \le LT \end{aligned} \tag{7}$$

In these equations, w_{jk} is the weight from feature coding unit *j* to motor coding unit *k*, the d_w weight decay rate ensures that only repeated co-activations result in stable weight learning, *LR* denotes the learning rate (i.e., the magnitude of the change in weights for each learning trial), $Act_j(t)$ is a value based on the activation of feature coding unit *j*, $Act_k(t)$ is a value based on the activation of motor coding unit *k*, *LT* is the learning threshold (above which the activation levels of both units must be in order to engage in weight learning) and $A_j(t)$ and $A_k(t)$ are the actual activation levels at time *t* of feature coding unit *j* and motor coding unit *k* respectively. Note that we rescale the activation of both units to their respective proportion to the learning threshold and that the computed connection weights are bound to vary between 0.0 and 1.0.

The total number of codes (coding units) and connections varies with the specific instances of HiTEC used for the different simulations. All parameters and default values as used in the simulations are listed in the Appendix. In sum, these modeling equations and parameters allow for a biologically plausible simulation of activation propagation through a network of units. Higher decay rates make units decay faster; lower decay rates keep units very active for a longer period of time. Higher input values for external input and stronger weights between units result in faster activation propagation. Higher voltage thresholds make unit activation to a lesser extent enhanced by top down input; conversely, lower voltage thresholds lead to earlier and stronger influence of top down modulation on unit activation. Stronger weights between excitatory and inhibitory units strengthen the lateral inhibition mechanism. As a result, they reduce the time required to settle the competition between the units within a shared layer, after which only one unit remains strongly activated. Lower weights, conversely, lengthen this time to convergence.

Note that our ambition for HiTEC has not been to search for specific parameter values (e.g., thresholds, weight ranges and scaling parameters) in order to optimally fit specific data distributions. We rather set out to provide a proof of principle as to how neurally plausible representations and connectivity may realize stimulus-response translation while addressing critical theoretical issues such as action control (Chapter 3), automaticity (Chapter 4) and coping with task context (Chapter 6).

Simulating behavioral studies

To model a behavioral study in HiTEC, a specific instance of the HiTEC model is constructed with layers, codes (coding units) and connections that match the stimulus, response, and task characteristics of the simulated experiment. Crucially, connections between feature codes and task codes are set to reflect the exact task instructions.

In each simulation there are two phases: first, action effects are learned, reflecting the period in which the participants get acquainted with the keypresses and their effects, which is commonly part of behavioral experiments. In this learning phase, we allow the model a set number of learning trials to acquire the associations between feature codes and motor codes. Note that when a motor code is executed, the changes in the environment (i.e., its action effects) are presented by supplying input to the sensory codes. Propagating activation towards feature codes allows the model to learn the feature code – motor code associations.

In the subsequent, experimental, phase the model is presented with various stimuli by supplying input to specific sensory coding units. Gradually, activation spreads across all the involved coding units in the various network layers. The trial is terminated at the selection of a motor response and the reaction time is determined based on the number of cycles between stimulus onset and response selection. This enables comparing simulated reaction times with reaction times of human participants in behavioral experiments.

In each simulation, multiple simulated subjects are generated based on the same HiTEC model instance. Although the layers and codes in the networks of these simulated subjects are identical, the noise in activation propagation of coding units is random, resulting in individual differences in performance, as reflected in both varying reaction times and error trials. To be able to use between-subjects designs, simulated subjects are assigned to different group conditions (and receive, for example, different task instructions or stimuli). Mean reaction times and standard deviations are computed for each simulated subject and each condition.

Model dynamics

As depicted in Figure 7, when a stimulus is presented to the model, activation propagates from sensory codes to feature codes, involving task codes, other feature codes and motor codes simultaneously. In the figure, an example trial (incongruent trial in the Simon task; see Chapter 4 for the specific HiTEC model instance and actual simulation results) is shown. From the first cycle on a high, right stimulus tone is presented by feeding external input to the sensory codes 'Auditory high' and 'Auditory right'. During the subsequent cycles their activation levels rise accordingly. Simultaneously, activation propagates towards feature codes. Until cycle 21, these are predominantly the feature codes (e.g., 'Right' and 'High') connected to the active sensory codes.

Due to prior action-effect learning, feature code 'Right' propagates activation to motor code 'M_2', of which the activation level is rising during cycles 6 to 29. At the same time, activation propagates from the 'High' feature code towards the task codes, resulting in a relatively more strongly activated 'T_1' and less strongly activated 'T_2' from cycle 7 on. 'T_1' further propagates activation towards feature code 'Left'. As a result, this feature code's activation level rises from cycle 7 on and exceeds the activation level of 'Right' at cycle 24. At the same time activation propagates from 'Right' to the associated motor code 'M_1' which exceeds the activation level of 'M_2' at cycle 34 and reaches the response threshold at cycle 41. At that point, also feature codes 'Left' and 'Key' are highly activated.

Note that these feature codes resemble the action effect of the produced response. Also note that when 'M_2' would have been slightly more activated, this code could have reached the response threshold and the corresponding motor action could have been selected rather than 'M_1', constituting an error trial.

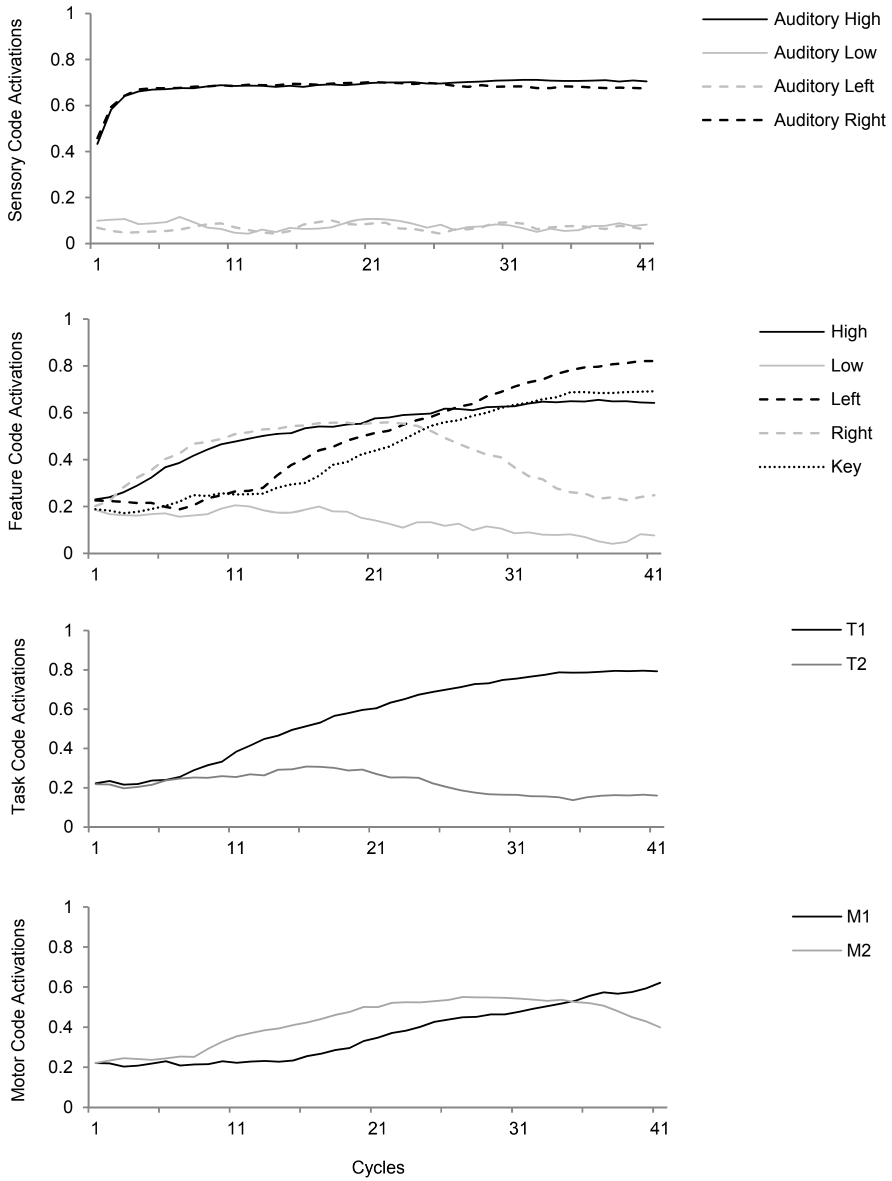

Figure 7. Interactive processing during a single stimulus-response translation trial (i.e., a high right auditory tone) involving representations at all levels simultaneously (example shown from Simon effect simulation. See Chapter 4 for more details).

Discussion

In HiTEC, neuron-like representations realize stimulus-response translation. Stimuli are presented by feeding external input to sensory codes. Responses are considered to execute when a motor code reaches the activation threshold. The connection between perception and action is realized by representations on multiple levels and interconnected by feedforward and feedback connections. The result is an integrated processing network that translates stimuli in responses by gradually propagating activation trough units in the model. Rather than a sequential stepwise process from sensory codes through intermediate representations to response codes, all representations at all levels cooperate and compete and together converge to a response outcome. Crucially, representations at higher levels modulate representations at lower levels. This allows both for direct interaction between perception and action representations and modulation by the task context.

Although the rather simple HiTEC model is not intended as a detailed neuroscientific model, it might be worth noting that its components, as well as their connectivity, do map in a gross way onto specific neural systems. The network architecture follows the same general form as more neurobiologically oriented models of visual attention and object selection do (e.g., Deco & Rolls, 2004).

Our approach is in line with the *integrated competition hypothesis* (Duncan, Humphreys, & Ward, 1997). This hypothesis proposes that visual attention results from competition in multiple brain systems and rests on the three following principles. First, different objects are considered to compete for activation within multiple brain systems. Second, although this competition takes place in multiple brain systems, it is integrated between these systems in such a way that units responding to the same object in different brain systems support each other's activity, whereas units responding to different object compete. Finally, competition is considered to be directed on the basis of relevant object properties based on the current task demands. Duncan et al. (1997) suggest top-down neural priming as a possible control mechanism. HiTEC could be considered both a generalization and specification of this hypothesis. Due to the common coding nature of feature codes, not only visual attention but also action anticipation (and thus action control) are considered to compete for activation, hence generalizing the scope of the integrated competition account. HiTEC further specifies a possible method of directing this competition using task set connections rather than priming. HiTEC explicitly addresses how the task instruction could implement such a task set and how task instruction could influence both perception and action planning.

The notion of interactive processing with mutual influences among multiple subsystems is shared by other models. For example, Ward (1999) proposes in his Selective Action Model that action plans may bias selective perceptual processing towards relevant objects. In his model, representations of a single object and its implications for actions are selected due to gradual and coordinated processing in multiple systems of perception and action. Similarly to HiTEC, the model aims at formulating an alternative to the sequential models of perception and action. To this end, the model follows the integrated competition hypothesis of visual attention and further integrates action systems. In similar fashion as HiTEC, selected representations receive external input and activation gradually spreads among various units coding through the reciprocal connections converging to a selected object and action. Task context is encoded by priming the units that represent the object feature (e.g., the color red) to look for or the action feature (e.g., a grabbing action) to execute. This biases the global competition resulting in response time differences between different conditions. The most important difference between Ward's (1999) model and HiTEC concerns the model architecture: Ward has explicitly taken the ventral 'what' and dorsal 'where' pathways (Milner & Goodale, 1995) into account resulting in two hardwired pathways between perceptual and action systems. HiTEC, in contrast, is based on TEC and thus contains a common coding level of feature representations that are used both for perception and action planning. Another major difference between the models is how a task is internalized. In Ward's model, a selection of codes receives a priming bias input. In this sense, stimulus presentation and task instruction occur simultaneously and using the same mechanism of applying external input. In HiTEC, in contrast, task context is internalized by interconnecting feature codes and generic task codes. These pathways subsequently modulate the propagation of activation resulting from stimulus presentation. Crucially, this allows HiTEC to internalize multiple task rules that compete during subsequent stimulus-response translation, whereas the Ward model seems to be confined to executing one specific task rule depending on the code(s) that receive additional input bias. Finally, although Ward, in accord with our approach, aimed at addressing the interaction between perception and action, his model assumes the implications for action of a given object by fixed connections between object features (e.g., vertical object orientation) and specific actions (e.g., vertical grasp). This connection between perception and action planning is addressed more explicitly in HiTEC using the notion of common codes and ideomotor learning (see Chapter 3). Moreover, these mechanisms allow addressing the issue of automaticity (see Chapter 4), which is not a matter of interest (or readily possible to account for) in the Ward model.

More recently, Botvinick et al. (Botvinick, Buxbaum, Bylsma, & Jax, 2009) further developed the Ward model. In their simulations, they explicitly link specific object features (e.g., color, shape, location) to specific response representations (e.g., reach actions, manual actions, color naming, respectively). In accord with Ward, they find that implementing a task set (i.e., priming specific actions) results in top down input to object features and, hence, in selective attention for objects having these features. Most points of comparison between HiTEC and the Ward model also apply here: the connections between object features and specific actions are assumed, the task set is implemented as additional input to action codes and automaticity is not addressed.

Summarizing, in addressing the interaction between perception and action these models extend (visual) attention for objects with a system that takes action features into account by means of reciprocal connections between perception and action subsystems. How these connections follow from experience or the task context, however, is not explicitly addressed. Moreover these models do not address empirical findings of automaticity (i.e., stimulus-response compatibility) which is key considering their implications for direct interaction between perception and action.

Well-known models of automaticity (e.g., Cohen, Dunbar, & McClelland, 1990; Kornblum et al., 1990; Kornblum, Stevens, Whipple, & Requin, 1999; Zorzi & Umilta, 1995) typically share the general (PDP) connectionist approach with units and excitatory and inhibitory connections. The model of the Simon effect by Zorzi and Umilta (1995), for example, contains stimulus feature codes and response codes. The stimulus feature codes propagate activation towards the response codes. The response codes compete for activation due to their mutual inhibitory connection. In contrast with the models described above, the connections between stimulus codes and response codes are one-directional. That is, stimulus codes activate response codes, not the other way around. In general, these models are more focused on the *process* of translating stimuli to responses and aim at fitting their simulation results to behavioral data. In this endeavor, Kornblum et al. (1999) explicitly divide processing in two distinct sequential stages: stimulus processing and response production. This division ensures that no processing takes place in the response-production stage until activation in the stimulus stage has reached threshold; this is in sharp contrast with HiTEC and the models of the interaction between perception and action discussed above. It does, however, allow them to fit their model to behavioral data on specific stimulus onset asynchrony effects in time courses in SRC effects. In contrast to these dual route process models, HiTEC and both the Ward and Botvinick et al. models take (neurally inspired) representations and reciprocal connectivity into account. The dual route models are discussed more elaborately in Chapter 4 where we discuss the topic of automation.

Finally, it must be stressed that HiTEC has a fairly simple architecture, modeling only a minimal basis of neuroscientific findings. The human brain has many more mechanisms known to mediate perception and action (e.g., subcortical structures such as the superior colliculus and the thalamus). In addition, processing in cortical areas is mediated by a variety of factors (e.g., neurotransmitters) and top down influences and lateral competition, central in HiTEC's interactive processing, are shown to await a first, fast feedforward sweep of activation in visual processing (Lamme & Roelfsema, 2000) suggesting distinct modes of vision and that assuming immediate interaction between multiple levels is rather simplified.

However, despite these simplifications, HiTEC's key assumptions – multiple level representations, common coding level, ideomotor learning, biased competition, reciprocal connections – lead to rather complex and interesting dynamics. We believe that these dynamics may shed light on the interaction and coordination of perception and action planning in human behavior. More specifically, we address how situation-specific meanings of actions emerge in action control (Chapter 3), how and why automaticity occurs (Chapter 4) and how task context may modulate perception and action planning in order to coordinate behavior (Chapter 5).