ARTICLE IN PRESS

Consciousness and Cognition xxx (xxxx) xxx–xxx

Contents lists available at ScienceDirect

Consciousness and Cognition

journal homepage: www.elsevier.com/locate/concog

Review article

Contingency and similarity in response selection

Wolfgang Prinz

Max Planck Institute for Human Cognitive and Brain Sciences, Stephanstrasse 1a, 04103 Leipzig, Germany

A R T I C L E I N F O

Keywords:
Choice reaction time tasks
Task representation
Event codes
Task sets
Similarity
Contingency

A B S T R A C T

This paper explores issues of task representation in choice reaction time tasks. How is it possible, and what does it take, *to represent such a task* in a way that enables a performer *to do the task* in line with the prescriptions entailed in the instructions? First, a framework for task representation is outlined which combines the implementation of task sets and their use for performance with different kinds of representational operations (pertaining to feature compounds for event codes and code assemblies for task sets, respectively). Then, in a second step, the framework is itself embedded in the bigger picture of the classical debate on the roles of contingency and similarity for the formation of associations. The final conclusion is that both principles are needed and that the operation of similarity at the level of task sets requires and presupposes the operation of contingency at the level of event codes.

1. Choice reaction time tasks

The topic of this paper is quite distant from Bruce Bridgeman's major fields of scientific interest and study. Still, the spirit of the argument aims to come close to Bruce's intellectual style and his way of approaching scientific problems. One of the key features of this style was to combine new experimental observations with old theory: use classical theory to understand novel findings – and, at the same time, use novel findings to elaborate on classical theory. This is exactly what this paper endeavors to do. Its topic is both novel and old. On the one hand, it is "novel" in the sense of addressing an experimental paradigm that has flourished over the past decades – the paradigm of choice reaction time tasks. On the other hand, it is "old" since it addresses theoretical issues that have been debated since the time of Aristotle – namely the roles and the workings of contingency and similarity in mental functioning. Following the spirit of Bruce's intellectual style, the argument will combine novel observations with old ideas, trying to find out how they can inform one another.

Since the time of Helmholtz, Donders, and Wundt choice reaction time tasks have provided an important tool for the study of human performance (Donders and Koster (1868)/1969; Helmholtz, 1867/1924; Wundt, 1902/1904). As research over the past decades has shown, these tasks allow to address the representational underpinnings of basic cognitive operations like stimulus identification, response selection, and response preparation (cf., e.g., Hick, 1952; Sanders, 1998; Smith, 1980; Sternberg, 1969; Welford, 1968, 1980). At the same time, they allow to address basic issues entailed in the formation of task representations as tools for guiding and controlling the interaction of these operations in accord with the demands of the task at hand (cf., e.g., Ach, 1905; Bunge, 2004; Kiesel et al., 2010; Koch, Gade, Schuch, & Philipp, 2010; Logan, van Zandt, Verbruggen, & Wagenmakers, 2014; Prinz, 2015; Schumacher & Hazeltine, 2016; Verbruggen, McLaren, & Chambers, 2014).

In what follows I examine choice reaction time tasks from a task representation perspective: how is it possible to *represent a task* in a way that enables performers to *do the task* according to the prescriptions provided by instructions? When we run experiments, we take it for granted that participants can do what the task requires them to do and we don't care about the miracle that their

E-mail address: prinz@cbs.mpg.de.

https://doi.org/10.1016/j.concog.2018.04.014
Received 11 January 2018; Received in revised form 23 April 2018; Accepted 24 April 2018
1053-8100/ © 2018 The Author. Published by Elsevier Inc. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/BY-NC-ND/4.0/).

Please cite this article as: Prinz, W., Consciousness and Cognition (2018), https://doi.org/10.1016/j.concog.2018.04.014

performance must somehow be guided and controlled by representations of the underlying task demands. This is the miracle I am addressing here.

Issues of task representation can be discussed at two levels, implementation and performance. The study of task implementation considers the operations involved in *setting the task.* Analysis at this level addresses the formation of task representations through instructions and practice trials. Instructions for choice tasks must specify three things: the set of possible stimuli, the set of possible responses, and the mapping rules for assigning responses to stimuli. For instance, a minimal choice task may require to map two stimuli (S_1, S_2) onto two responses (R_1, R_2) such that S_1 requires R_1 and S_2 requires R_2 as response. More complex tasks may come with larger sets that may differ in size (so that, for example, eight possible stimuli converge on four responses etc). Whatever the task may be, its implementation requires participants (i) to *understand the instructions* and the task demands entailed in it and (ii) to *form a task representation* that allows them to do the task. Practice trials help to shape this representation and improve its efficiency.

Conversely, the study of task performance considers the operations involved in *doing the task*. Analysis at this level addresses the utilization of representations for task execution. A typical choice reaction task consists of a sequence of trials. On each trial, a stimulus is presented, and participants are required to deliver the appropriate response as fast as they can. Each trial may thus be seen to entail an act of *stimulus-based response selection*. In other words, each trial requires to select one element from the response set, namely the particular response assigned to the current stimulus by the mapping rules specified in the instruction. The time that this act requires forms part of the reaction time, i.e. the total time that elapses between stimulus and response onset.

What does it take to implement a task representation for efficient response selection and what does it take to perform the task according to instructions? To address these questions I take two steps. The first concerns the nature of stimulus- and response representations. What does it take to represent stimuli and responses? What is same and what is different about representations of stimuli and responses? The second addresses the nature of task representations and the assignments entailed in them. What does it take to represent assignments between stimulus- and response representations and how can these assignments be used for action selection?

2. A Framework for response selection

To examine choice tasks, we may adopt two perspectives, external and internal. From an external point of view, we may consider an individual generating certain actions in response to certain stimuli presented to her, thereby following certain rules to which she has committed herself. For instance, she may respond to red vs. green stimulus patches by pressing left vs. right response keys, based on a previously acquired mapping rule that specifies color-to-key mappings.

Conversely, we may adopt an internal point of view. From this perspective, we invoke structural and functional features of a putative cognitive architecture that may account for the observed performance. For instance, as already indicated, we may invoke an architecture harboring stimulus codes, response codes, and linkages between them. Stimulus codes are understood as internal placeholders for external stimuli. They are generated and maintained on the sensory input side of the architecture. Response codes are understood as internal placeholders for external responses. They are generated and maintained on the motor output side. Finally, mappings between stimulus- and response codes are understood as internal placeholders for the stimulus-response assignments specified by task instructions.

2.1. Two principles

So far, the internal description does not provide much more than an internalized version of the external account. In order to make it work, we need to specify what stimulus- and response codes are, and how they are related to each other. Here we encounter two basic principles, separate and common coding (Hommel, Müsseler, Aschersleben, & Prinz, 2001; Prinz, 1984, 1990, 1997).

Separate coding – Separate coding reflects the view that stimuli and responses are entirely different and incommensurate things that have nothing to do with each other. Stimuli are things and events that happen in the outside world, whereas responses are events generated by the body's motor system. Hence, their representations, or codes, must be entirely disjunct and incommensurate. Stimulus codes must carry information about the things and events that have given rise to their formation, and response codes must carry information about the movements to which they give rise. Thus, while stimulus codes specify sensory features like hues or pitches, response codes specify motor features like muscles and forces. If so, there can be no common denominator for them. Still, in order to account for the relationships between stimuli and responses in overt behavior, we need ways of bridging the gap between them. An obvious way to get there is to posit associative linkages that connect one to the other.

Given psychology's historical record of explaining behavior and performance in terms of associations between stimuli and responses, it is perhaps not surprising that separate coding is the mainstream approach in the field (cf., e.g., Broadbent, 1958; Posner, 1978; Sanders, 1998; Sternberg, 1969; Welford, 1968, 1980). Its power is rooted in its universality. Since associations can in principle link anything to anything, they offer themselves as ideal candidates for implementing mapping rules as sets of arbitrary linkages between stimulus- and response codes. Occasionally, the metaphor of *translation* has been used to characterize the logic of the underlying mechanism (e.g., Sternberg, 1969; Welford, 1960, 1968; cf. Prinz, 1990, pp. 169/170). This metaphor suggests that stimuli may specify responses in virtue of linkages between codes from two separate and disjunct coding systems. These systems work like two languages so that something like a dictionary is required to translate one into the other.

Common coding – In one sense, separate coding must thus be true. Still, there are reasons to believe that it doesn't reflect the full truth. A major challenge comes from so-called privileged linkages between perception and action, claiming for a functional role of similarity in the underlying mechanism. The crucial observation here is that stimuli can often more easily be linked to responses resembling them as compared to non-resembling ones. For instance, in the Simon task a stimulus patch on the left side of the screen can be more easily and more efficiently linked to a key press on the left side than on the right side (Simon, 1969). The same holds true of a large number of further instances of so-called stimulus-response compatibility effects and a rich variety of instances of action-induced movements (such as, for example, imitation, rhythmic entrainment, ideomotor movements etc.; for overviews see Hommel et al., 2001; Hommel & Prinz, 1996; Prinz, 1990; Proctor & Reeve, 1990).

Separate coding has no obvious way to account for such similarity-based linkages since it regards stimulus and response codes as incommensurate. These incommensurate codes do not share any features on which similarity-based operations could be grounded. This is where the Theory of Event Coding (TEC) and the notion of common coding come into play, suggesting to extend the classical framework in a way that posits, besides arbitrary linkages between incommensurate codes, similarity-based linkages between commensurate codes as well (Hommel et al., 2001; Prinz, 1990, 1997). Occasionally, the metaphor of *resonance* has been used to characterize the way in which stimuli may induce responses in a common coding architecture (e.g., Koffka, 1924). This metaphor suggests that stimuli may specify responses in virtue of shared content. Put in a more functional language, the claim is that stimulus codes share features with response codes so that, when one of them gets activated, the other starts resonating with it. Accordingly, no translation and no dictionary is required to get from one to the other.

Two systems? – Taken together, the evidence seems to suggest that a full account of stimulus-based response selection requires to combine the two principles. One way of getting there is to invoke two systems operating in parallel – one based on linkages between separate, incommensurate codes and the other based on overlap between common, commensurate codes. Two-systems, or dual-route accounts along this line seem to be a natural answer to the challenge provided by two competing principles. Typical dual-route accounts invoke (i) a default system for arbitrary S-R assignments, (ii) a fast-track system for overlapping stimulus- and response codes and (iii) a race between the two systems as they operate in parallel (Kornblum, Hasbroucq, & Osman, 1990; see also Coltheart, Rastle, Perry, Langdon, & Ziegler, 2001).

Still, one may doubt whether this is the only way to combine the two principles. To account for two principles in terms of two independent systems does only make sense if the operations instantiating them are independent from one another. However, this is not always the case. In classical compatibility paradigms the functional condition of code overlap is often not given a priori but created a posteriori so that common coding at one level of representation is made possible through separate coding and arbitrary connections at another level. The Stroop effect provides an obvious example. On the one hand, it is fairly obvious that the visual shape of the word GREEN does not share any features with the motor program for articulating it, nor with the auditory stimulus pattern arising from pronouncing it. Still, the three share a common code – at least for those individuals who look back on a long history of learning the underlying grapheme/phoneme correspondence rules. For those who have learned these arbitrary rules, the visual stimulus overlaps, as it were, with the vocal response and the auditory stimulus. Lifelong learning has established linkages that connect the visual shape of the word with the auditory pattern of its name and the articulatory pattern of the corresponding vocal response. If so, overlap in the common domain of name representations is made possible through linkages with entries in the separate domains for visual, auditory, and articulatory representations. Below, I come back to this point in more detail.

2.2. Two levels

Two-systems frameworks cannot account for such interactions between separate and common coding. What is needed, instead, is a more integrated framework that allows to combine the two principles. Here, I give a brief sketch of a framework that considers the underpinnings of response selection at two interacting levels: event codes and task sets (see Prinz, 2015 for more detailed elaboration and application to various kinds of interference paradigms).

Event codes – The Theory of Event Coding (TEC) regards task sets as assemblies of event codes that are organized in specific ways (Hommel et al., 2001; Memelink & Hommel, 2005, 2006). In order to understand how these assemblies work, we first need to understand what event codes are and how they interact with each other.

The basic idea is simple (and perhaps a bit trivial). Event codes are cognitive representations of all kinds of things and events that mental activity may address. To fulfill their representational function, event codes need to be mutually commensurate, even if the things and events they stand for are incommensurate. How can this be possible? For instance, how can such diverse things like a color patch appearing on a computer screen and a participant's hand pushing a response key be represented in a common format? As has been shown elsewhere, distal reference is a key feature here. Event codes are commensurate in virtue of the fact that they refer to distal events in the world and body. They are, in other words, placeholders for external events outside the representing system.

A convenient way of conceiving the structure of event codes is to regard them as transient feature compounds that are created from features supplied from a permanent feature space (cf., e.g., Goldstone, 1994; Hommel et al., 2001; Tversky, 1977). For instance, the color patch on the screen may be represented by a compound of features such as *red/ground/small/at top of screen* etc. These features are semantic in the weak sense of providing elementary building blocks for meaningful event codes.

Let us take these building blocks for granted and examine how event codes can be formed from them. Fig. 1 provides a simple illustration. Panel (A) shows a hypothetical feature compound which is characterized through a particular set of features and a particular pattern of connectivity interlinking them. A simple structure like this can undergo various kinds of changes and elaborations. For instance, as shown in panel (B), features may differ in their relative contributions to an event code. While some may be essential and defining, others may be more or less accessory. For instance, the color of the patch may be crucial for the task at hand whereas its shape may be irrelevant. These demands modulate the relative weights with which features contribute to their event codes. A more radical way of altering the structure of event codes is to include new and/or delete old features, as shown in panel (C). These alterations furnish feature compounds with an enormous structural flexibility: the capacity to grow and shrink on demand. As

ARTICLE IN PRESS

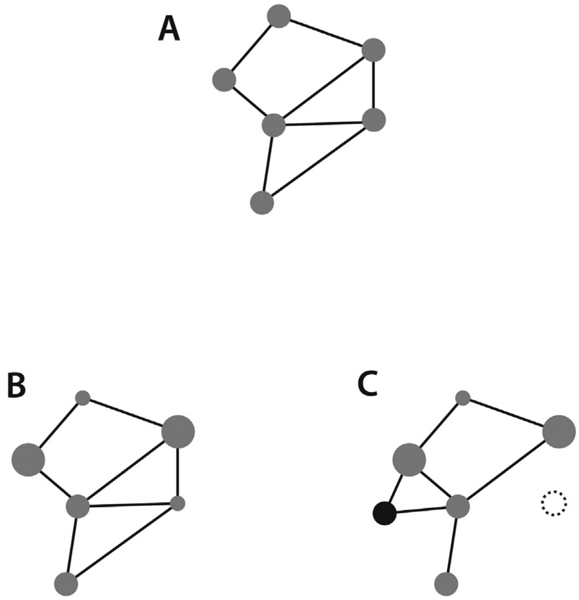

Fig. 1. Three feature compounds. (A) Compound with equal weights for all features; (B) Same compound with different weights (indicated by different node sizes); (C) Same compound with one of the original features deleted (dotted) and a novel feature included (black). *Reprinted from Prinz W (2015) Task representation in individual and joint settings. Front. Hum. Neurosci. 9:268. doi: https://doi.org//10.3389/fnhum.2015.00268, p. 4.*

we will see below, this capacity turns out to be of utmost importance for tailoring the structure of event codes to the demands of the task at hand.

How can such changes come about? The basic claim is that codes that are involved in the representation of a given task become selectively tuned to the needs of efficient interaction with other codes involved in the same task. To illustrate, consider Fig. 2. Panel (A) shows two intersecting event codes, a red one and a blue one. Each of them exhibits both shared and unshared features. While shared features are common to both codes, unshared features are specific to each of them. Such overlap may either be beneficial or detrimental, depending on the kind of interaction in which the two codes are involved. Overlap will be beneficial in task scenarios requiring both codes to become jointly activated at the same time (*code cooperation*). By contrast, it will be detrimental in scenarios requiring one code becoming active and the other silent at the same time (*code competition*). While cooperation requires that activating one of them co-activates the other, competition requires that they are shielded from such mutual co-activation.

The structure depicted in panel (A) invites obvious measures for improving the efficiency of code interaction. Panels (B) and (C) show measures supporting code cooperation, such as increasing feature weights or creating new features in the overlap zone and/or weakening weights and deleting unshared features in the non-overlap zone. By contrast, panels (D) and (E) show measures supporting code competition. Again, they may either pertain to the overlap zone (weakening or deleting shared features) or the non-overlap zone (strengthening or creating unshared, distinctive features). – Taken together, the common principle underlying these alterations is that the required mode of code interaction acts back on code structure, strengthening and weakening shared and unshared features on demand.

Task sets – The application of these ideas to task set formation is fairly straightforward. Our framework claims that event codes provide the basic equipment from which task sets are made. It regards, in other words, task sets as particular kinds of assemblies of such codes. The structure of these assemblies reflects, on the one hand, the history of code interactions involved in foregoing task implementation, and it predicts, on the other hand, the pattern of code interactions involved in subsequent task performance. While implementation tailors content profiles of stimulus- and response codes to the demands of the task, performance reflects the pattern of code interactions to which the implemented task set gives rise (Memelink & Hommel, 2005, 2006).

On each trial, a choice task requires two basic operations: identification of the stimulus event and selection of the appropriate action for response. When considered in terms of event code assemblies, each of these operations is seen to entail *selective competition*, i.e. the selection of one code from a set of competing codes. Consider, for instance, a task in which red vs. green stimuli require left vs right responses. In terms of the scheme depicted in panel (A) of Fig. 3, such selective competition requires horizontal interactions *among* stimulus- and response codes, respectively. Stimulus identification must select one particular stimulus code from the set of competing stimulus codes. Likewise, response selection must select one particular response code from the set of competing response codes.

As indicated above, selective competition requires tailoring the involved event codes and their assemblies according to the demands of the task. An obvious way of getting there is to strengthen distinctiveness through weakening code overlap. This is shown in panel (B) where overlap is deleted among both stimulus codes and the competing response codes. As a result of such distinctive tuning, conflict is reduced and the efficiency of selection increases. Importantly, this applies to both stimulus identification and response selection.

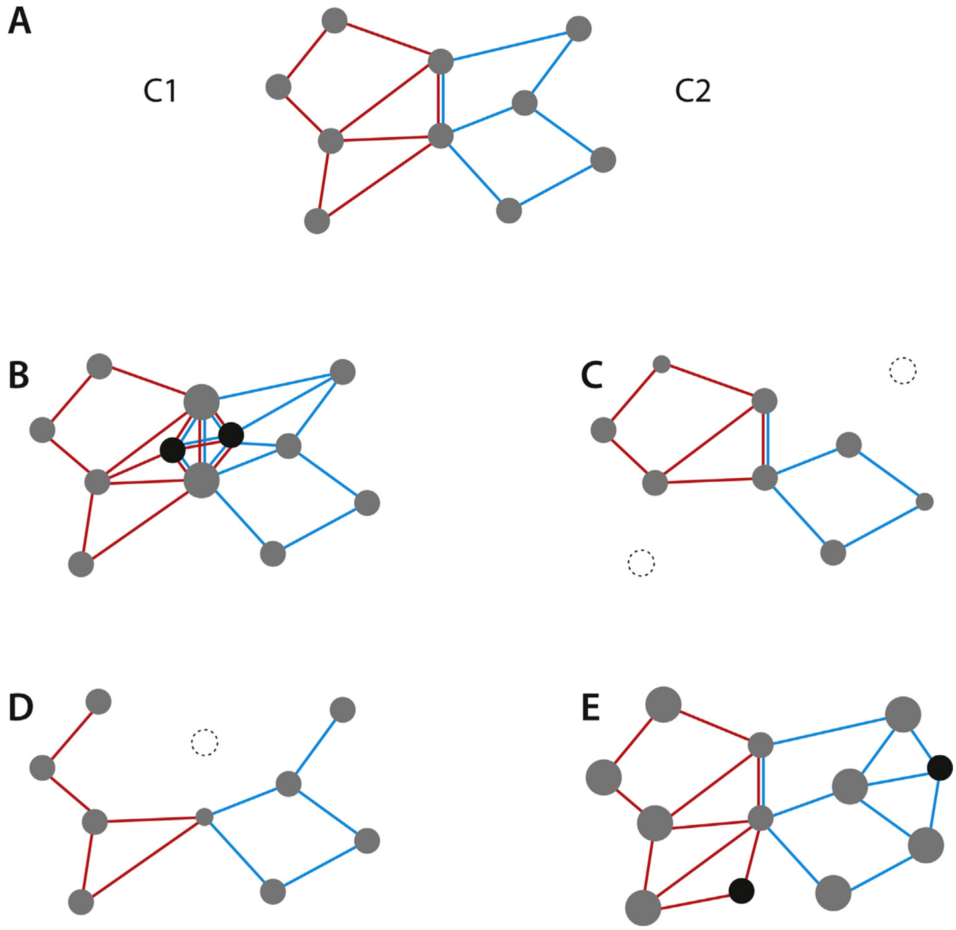

Fig. 2. Two overlapping event codes, instantiated as intersecting feature compounds (C_1 and C_2 with red and blue edges, respectively). Overlap is indicated by feature nodes on which red and blue edges converge (= shared features). (A) Initial scheme with equal feature weights; (B/C) Possible changes under conditions of code cooperation: (B) Strengthening shared features through (i) increasing weights of old features (node size) and (ii) including new shared features (black); (C) Weakening unshared features through weight reduction and deletion (dotted); (D/E) Possible changes under conditions of code competition; (D) Weakening shared features through weight reduction and deletion; (E) Strengthening unshared features through weight increase and inclusion of novel unshared features (black). *Reprinted from Prinz W (2015) Task representation in individual and joint settings. Front. Hum. Neurosci. 9:268. doi: https://doi.org//10.3389/fnhum.2015.00268, p. 5.* (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Mapping rules require that activation of particular stimulus codes must be followed by activation of particular response codes. In order to instantiate these rules, the involved event code assemblies need to establish what we may call *selective cooperation* between stimulus- and response codes. With reference to the scheme depicted in panel (B) of Fig. 3, cooperation requires vertical interactions *between* stimulus- and response codes. These interactions may take two complementary forms, aiming at facilitating wanted and impeding unwanted mappings, respectively. Notably, the distinction between these two kinds of mappings is grounded on instructions: wanted mappings yield correct responses whereas unwanted mappings yield wrong responses. Efficient task sets thus require both to facilitate wanted and impede unwanted mappings.

Facilitation of *wanted mappings* requires selective cooperation between particular stimulus- and response codes. An obvious way of instantiating such cooperation is to create selective overlap between the involved codes. When the task requires pressing a left vs. right key in response to a red vs. green color patch, the required mappings are initially entirely arbitrary. Before the task is administered, no linkages exist between such features as *red – left*, or *green – right*. This is where instructions come in, acting to create patterns of selective overlap that instantiate these mappings.

As panel (B) of Fig. 3 illustrates, we may discern two strategies of creating such overlap. One is to integrate given stimulus features into response codes and/or given response features into stimulus codes. As a result, *red* becomes integrated into the response code for *left* key presses and/or *left* becomes integrated into the stimulus code for *red* patches. The other way is to generate entirely new features and assign them to each of the two codes that are required to cooperate. As a result, a new feature becomes integrated into both the stimulus code for *red* patches and the response code for *left* key presses, thus instantiating the wanted mapping between the two codes (as well as the external events they stand for).

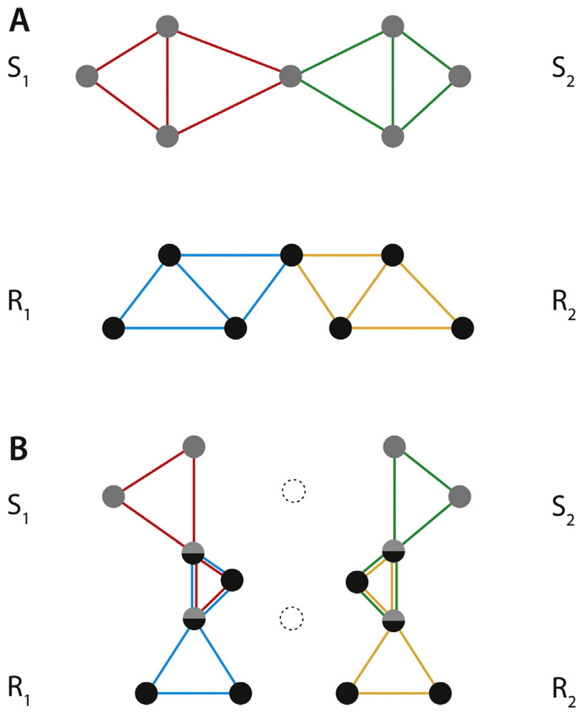

Fig. 3. Interactions between event codes in task set formation. (A) Initial structure of a hypothetical event code assembly, with two partially overlapping stimulus codes (S_1/S_2) and two partially overlapping response codes (R_1/R_2). (B) Final structure of the same assembly with initial horizontal overlap *among* stimulus and response codes deleted (unconnected dotted features) and new vertical overlap *between* stimulus and response codes created (either by mutual integration of old features (black/grey) or generation of entirely novel features (black). Note that the final structure of the assembly differentiates between wanted and unwanted S-R assignments (wanted: S_1-R_1 and S_2-R_2; unwanted: S_1-R_2 and S_2-R_1). *Modified after Prinz W (2015) Task representation in individual and joint settings. Front. Hum. Neurosci. 9:268. doi: https://doi.org//10.3389/fnhum.2015.00268, p. 6.*

Such feature assignments may at first glance appear fairly strange. Yet, we need to remind ourselves that features are semantic free-floaters and not naturally tied to the stimulus or the response domain. Each feature can therefore become integrated into each kind of code. Furthermore, we need to remind ourselves that features are semantic elements that may not only address physical properties (such as colors or locations) but also other kinds of functional or relational properties (such as locations *belonging to* colors, or colors *belonging to* locations etc.). As a result, such functional features instantiate the normative fact that, based on the mapping rules entailed in the instruction, the two events have to be assigned to each other in the current task scenario. To support co-operation in line with such wanted assignments, the respective codes create new overlap, thus using code similarity as a means for instantiating arbitrary mapping rules[1]. Related ideas about relationships between rule-based and similarity-based coding have recently been advanced in systems neuroscience as well (see Kriegeskorte, Mur, & Bandettini, 2008).

While selective generation and strengthening of new overlap aims at facilitating wanted mappings, selective deletion and weakening of old overlap aims at impeding *unwanted mappings*. Selective deletion comes into effect under conditions of unwanted shared features, that is, when feature overlap arises between stimulus- and response codes that are *not* assigned to each other so that their unwanted cooperation would drive erroneous responses. A simple example is provided by a binary choice task with incompatible assignments between stimuli and responses. For instance, when the task requires *right* vs. *left key presses* in response to tones presented to the *left* vs. *right ear*, unwanted assignments are (spatially) compatible and need to be weakened (whereas wanted but incompatible assignments need to be strengthened).

To sum up, this framework views task sets as code assemblies in which stimulus- and response codes interact with each other and shape each other. On the one hand, they compete with each other *within* their respective sets, to the effect of weakening and deleting overlap *among* stimulus codes and among response codes (*selective competition)*. On the other hand, codes cooperate *across* their respective sets, to the effect of strengthening and weakening overlap *between* stimulus and response codes *(selective cooperation)*. Strengthening facilitates wanted assignment whereas weakening impedes unwanted assignments. These code interactions serve to implement the task in a way that tailors the underlying code assembly to the needs of efficient performance.

Task implementation and task performance thus rely on operations at different levels of task representation. As concerns

[1] As an anonymous reviewer pointed out, a mechanism along these lines – i.e. creation of free-floating functional features instantiating relationships between other features – may also help providing some understanding of other perceptual phenomena like, e.g., illusory conjunctions (Treisman & Schmidt, 1982; see also Bridgeman, 1989) or grapheme-color synesthesia (Brang, Rouw, Ramachandran, & Coulson, 2011; see also Bridgeman, Winter, & Tseng, 2010).

implementation, the critical operations on which the formation of task sets is grounded pertain to structure and composition of the *feature compounds* that make up individual event codes (such as adding new/subtracting old features or altering their weights in their respective compounds). Since these compounds are networks of features (nodes) and linkages (edges) that link any features to any others on demand, their structure mainly reflects the contingencies of feature co-occurrence that the task requires. As concerns performance, the critical operations on which the act of response selection relies pertain to code interactions within the *event code assembly* that makes up the task set for the task at hand (such as assessing and comparing overlap between the involved stimulus- and response codes). Since these assemblies are conceived as transient groupings of free-floating event codes in a common representational domain, their structure mainly reflects relationships in terms of code overlap, or similarity. At this level, stimulus- and response codes are commensurate so that the resulting pattern of code overlaps reflects the pattern of similarities of their contents.

3. Contingency and similarity

So what we have here are two kinds of operation that rely on different principles of representational interaction. Contingency-based feature compounds are created and shaped at one level to enable similarity-based code interactions at another level (formation of feature compounds and event code assemblies, respectively). According to this scheme, the operation of similarity requires, and in fact presupposes, the foregoing operation of contingency: performance can draw on similarity-based interactions between stimulus- and response codes *if and only if* implementation has before provided contingency-based operations for the task-specific tuning of the underlying feature compounds.

A two-level scheme like this raises the issue what the functional benefits of similarity-based processing may be. What can similarity-based code matching do for a cognitive system that contiguity-based mapping cannot? Matching relies on overlap of intrinsic content, whereas mapping requires scaffolds of previously established extrinsic linkages. Matching systems can therefore be more flexible than mapping systems. Most importantly, they continue to work when pre-established scaffolds do not exist or break down, for instance when novel, unfamiliar stimuli are presented and/or new responses are required (Prinz, 1984, 1990). This seems to be the crucial asset for which cognitive systems invest substantial resources when they construct devices for common coding and similarity-based matching. Still, we must not forget that similarity-based matching at the level of code assemblies is dependent on the working of an underlying machinery for contingency-based mapping at the level of feature compounds.

To conclude, let us see how our local framework fits into the bigger picture of global principles of mental functioning. The bigger picture addresses the classical debate on principles underlying relationships and interactions between ideas and representations - so-called *Gesetze des Vorstellungsverlaufs,* as a conventional German term goes (cf., e.g., Wundt, 1898).

What are the principles that determine how ideas or representations are related to and follow each other? Over centuries, if not millennia, two basic answers have been struggling against each other, resorting to extrinsic context and intrinsic content, respectively. The first claims that ideas in the mind become related to each other in virtue of contingency, that is, in virtue of joint occurrence in *common context.* As a result, they link with each other, with linkage strength reflecting the frequency/reliability of their joint occurrence. The resulting pattern of linkages then reflects the way in which ideas map onto each other, thereby controlling how one translates into the other. The second answer claims that ideas become related to each other in virtue of similarity, or *common content.* As a result, they overlap with each other, with amount of overlap reflecting their similarity. The pattern of overlaps then reflects the way in which ideas match each other, thereby controlling how they resonate with each other. As mentioned above, the metaphors of translation and resonance have often been used to address contingency- and similarity-based relationships between ideas and representations.

Our framework applies these general principles of representational interaction to the special domain of response selection, that is, to transitions from stimulus to response codes. For making these transitions it invokes both common coding (based on overlap of intrinsic content) and separate coding (based on linkages as extrinsic scaffolds). It resorts to common coding and intrinsic overlap at the level of code assemblies, but to separate coding and extrinsic linkages at the level of feature compounds, claiming that the operation of similarity (at the code assembly level) builds on the foregoing operation of contingency (at the feature compound level).

When held against the classical debate on principles, this claim seems to be at odds with related claims raised in earlier rounds of that debate. A famous example is Höffding's discussion of association and recognition about 130 years ago (Höffding, 1889/1890; see also Köhler, 1973; Koffka, 1935; Neisser, 1967). In discussing laws of association, Höffding argued that contingency-based associations (such as *bread – butter*) cannot be understood without taking foregoing similarity-based operations into account that take care of the recognition of the stimulus. It may be true, Höffding argued, that the path of association which leads from *bread* to *butter* relies on contingency-based translation. However, there is another path involved in the task that is usually overlooked: the path of recognition that leads from the transient stimulus to the permanent representation, i.e. from the stimulus word presented on the screen to the concept stored in memory. Höffding argued that this path (which became later named as *Höffding's step*) must rely on similarity-based resonance.

So, while Höffding claimed that similarity must precede contingency, our claim is that contingency must precede similarity. However, on a closer look, the seeming contradiction goes away. This is because Höffding's argument addresses two different stages of processing (i.e., recognition and subsequent association), whereas our argument addresses two different levels of representation (i.e., feature compounds and code assemblies). Accordingly, they do not exclude each other. An obvious way of reconciling them is to assume that the above-raised claims for response selection must likewise apply to stimulus recognition (i.e., Höffding's step). For this step, too, it must apply that similarity-based operations at the code level require and presuppose contingency-based operations at the feature level that act to shape feature compounds in a way that allows for common coding at the code level.

If this is true, we must conclude that contingency comes first and similarity second. Contingency must be considered the primary principle underlying relationships and interactions between ideas and representations. Contingency builds on separate coding, creating networks of feature nodes and linkages. This principle is universal and ubiquitous. It builds on Hebbian learning, linking anything to anything (Hebb, 1949). However occasionally, contingency and separate coding at one level may, in the interest of flexibility, give rise to the formation of representational structures that allow for common coding and a role of similarity at another level. However, since it builds on overlap of content, this principle is much more selective and cannot link anything to anything. Similarity must thus be considered a secondary, less universal and less ubiquitous principle of mental functioning.

References

Ach, N. (1905). *Über die Willenstätigkeit und das Denken.* Göttingen: Vandenhoeck & Ruprecht.
Bunge, S. A. (2004). How we use rules to select actions: A review of evidence from cognitive neuroscience. *Cognitive, Affective, & Behavioral Neuroscience, 4*(4), 564–579.
Brang, D., Rouw, R., Ramachandran, V. S., & Coulson, S. (2011). Similarily shaped letters evoke similar colors in grapheme–color synesthesia. *Neuropsychologia, 49*(5), 1355–1358.
Bridgeman, B. (1989). Modeling separate processing pathways for spatial and object vision. *Behavioral and Brain Sciences, 12*(3), 398.
Bridgeman, B., Winter, D., & Tseng, P. (2010). Dynamic phenomenology of grapheme-color synesthesia. *Perception, 39*(5), 671–676.
Broadbent, D. E. (1958). *Perception and communication.* Oxford: Pergamon Press.
Coltheart, M., Rastle, K., Perry, C., Langdon, R., & Ziegler, J. (2001). DRC: A dual route cascaded model of visual word recognition and reading aloud. *Psychological Review, 108*(1), 204–256.
Donders, F. C., & Koster, W. G. (1868/1969). On the speed of mental processes. *Acta Psychologica, 30*, 412–431.
Goldstone, R. L. (1994). Similarity, interactive activation, and mapping. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 20*, 3–28.
Hebb, D. O. (1949). *The organization of behavior.* New York: Wiley.
Helmholtz, H. V. (1867/1924). Helmholtz's treatise on physiological optics (J. P. C. Southall, Trans.). New York: Optical Society of America.
Hick, W. E. (1952). On the rate of gain of information. *Quarterly Journal of Experimental Psychology, 4*(1), 11–26.
Höffding, H. (1889/1890). Über Wiedererkennen, Association und psychologische Aktivität. Vierteljahresschrift für Wissenschaftliche Philosophie, 13, 420–458; 14, 427–454, 167–205, 293–316.
Hommel, B., Müsseler, J., Aschersleben, G., & Prinz, W. (2001). The theory of event coding (TEC): A framework for perception and action planning. *Behavioral and Brain Sciences, 24*, 849–878.
Hommel, B., & Prinz, W. (Eds.). (1996). Theoretical issues in stimulus-response compatibility (Vol. 118). Amsterdam: North-Holland.
Kiesel, A., Steinhauser, M., Wendt, M., Falkenstein, M., Jost, K., Philipp, A. M., & Koch, I. (2010). Control and interference in task switching – A review. *Psychological Bulletin, 136*(5), 849–874. http://dx.doi.org/10.1037/a0019842.
Koch, I., Gade, M., Schuch, M., & Philipp, A. M. (2010). The role of inhibition in task switching: A review. *Psychonomic Bulletin & Review, 17*(1), 1–14. http://dx.doi.org/10.3758/PBR.17.1.1.
Köhler, W. (1973). *Dynamics in psychology.* New York: Liveright.
Koffka, K. (1924). The growth of the mind: An introduction to child-psychology (R. M. Ogden, Trans.). New York: Harcourt, Brace & Co.
Koffka, K. (1935). *Principles of gestalt psychology.* London: Routledge.
Kornblum, S., Hasbroucq, T., & Osman, A. (1990). Dimensional overlap: Cognitive basis for stimulus-response compatibility: A model and taxonomy. *Psychological Review, 97*, 253–270.
Kriegeskorte, N., Mur, M., & Bandettini, P. (2008). Representational similarity analysis – Connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience, 2*, 4. http://dx.doi.org/10.3389/neuro.06.004.2008.
Logan, G. D., Van Zandt, T., Verbruggen, F., & Wagenmakers, E.-J. (2014). On the ability to inhibit thought and action: General and special theories of an act of control. *Psychological Review, 121*, 66–95. http://dx.doi.org/10.1037/a0035230.
Memelink, J., & Hommel, B. (2005). Attention, instruction, and response representation. *European Journal of Cognitive Psychology, 17*, 674–685.
Memelink, J., & Hommel, B. (2006). Tailoring perception and action to the task at hand. *European Journal of Cognitive Psychology, 18*, 579–592.
Neisser, U. (1967). *Cognitive psychology.* New York: Appleton-Century-Crofts.
Posner, M. I. (1978). *Chronometric explorations of mind: the third Paul M. Fitts lectures, delivered at the University of Michigan, September 1976.* Hillsdale, NJ: Erlbaum.
Prinz, W. (1984). Modes of linkage between perception and action. In W. Prinz, & A. F. Sanders (Eds.). *Cognition and motor processes* (pp. 185–193). Berlin: Springer.
Prinz, W. (1990). A common coding approach to perception and action. In O. Neumann, & W. Prinz (Eds.). *Relationships between perception and action: Current approaches* (pp. 167–201). Berlin: Springer.
Prinz, W. (1997). Perception and action planning. *European Journal of Cognitive Psychology, 9*(2), 129–154. http://dx.doi.org/10.1080/713752551.
Prinz, W. (2015). Task representation in individual and joint settings. *Frontiers in Human Neuroscience, 9*. http://dx.doi.org/10.3389/fnhum.2015.00268.
Proctor, R. W., & Reeve, T. G. (Eds.). (1990). Stimulus-response compatibility: An integrated perspective (Vol. 65). Amsterdam: North-Holland.
Sanders, A. F. (1998). *Elements of human performance: Reaction processes and attention in human skill.* Mahwah, NJ: Erlbaum.
Schumacher, E. H., & Hazeltine, E. (2016). Hierarchical task representation: Task files and response selection. *Current Directions in Psychological Science, 25*(6), 449–454.
Simon, J. R. (1969). Reactions toward the source of stimulation. *Journal of Experimental Psychology, 81*(1), 174–176.
Smith, G. A. (1980). Models of choice reaction time. In A. T. Welford (Ed.). *Reaction times* (pp. 173–214). London: Academic Press.
Sternberg, S. (1969). The discovery of processing stages: Extensions of Donders' method. *Acta Psychologica, 30*, 276–315.
Treisman, A., & Schmidt, H. (1982). Illusory conjunctions in the perception of objects. *Cognitive Psychology, 14*(1), 107–141.
Tversky, A. (1977). Features of similarity. *Psychological Review, 84*, 327–352.
Verbruggen, F., McLaren, I. P. L., & Chambers, C. D. (2014). Banishing the control homunculi in studies of action control and behavior change. *Perspectives on Psychological Science, 9*, 497–524. http://dx.doi.org/10.1177/1745691614526414.
Welford, A. T. (1960). The measurement of sensory-motor performance: Survey and reappraisal of twelve years' progress. *Ergonomics, 3*, 189–230.
Welford, A. T. (1968). *Fundamentals of skill.* London: Methuen.
Welford, A. T. (Ed.). (1980). *Reaction times*. London: Academic Press.
Wundt, W. (1898). *Grundriss der Psychologie.* Leipzig: Engelmann.
Wundt, W. (1902/1904). Principles of physiological psychology (E. B. Titchener, Trans.). London: Swan Sonnenschein; New York: MacMillan.