Systems Factorial Technology with R

Joseph W. Houpt
Wright State University, Dayton, Ohio

Leslie M. Blaha, John P. McIntire and Paul R. Havig
U.S. Air Force Research Laboratory, Wright-Patterson Air Force Base, Ohio

James T. Townsend
Indiana University, Bloomington, Indiana

Systems Factorial Technology (SFT) comprises a set of powerful nonparametric models and measures, together with a theory-driven experiment methodology termed the Double Factorial Paradigm (DFP), for assessing the cognitive information processing mechanisms supporting the processing of multiple sources of information in a given task (Townsend & Nozawa, 1995). We provide an overview of the model-based measures of SFT together with a tutorial on designing a DFP experiment to take advantage of all SFT measures in a single experiment. Illustrative examples are given to highlight the breadth of applicability of these techniques across psychology. We further introduce and demonstrate a new package for performing SFT analyses using R for Statistical Computing.

Introduction

Systems Factorial Technology (SFT) is a framework for studying how different sources of information combine in cognitive processing (Townsend & Nozawa, 1995). These sources can be as similar as visual information from the left and right visual field or as disparate as the demands of two different tasks such as driving while talking on a cell phone. SFT stands out as a particularly powerful framework because the various ways in which information can be combined are classified based on mathematically defined model properties. Despite the constraints due to the rigor of the definitions, SFT is quite general in that it requires no distributional or parametric assumptions about the cognitive processes. Using these precise mathematical definitions, there are a number of tests within the SFT framework to reject large classes of possible processing properties and support very specific properties.

While the mathematical rigor is an advantage of the framework, the technical details can be overwhelming. In this paper, we describe the general process of SFT experimental design and analysis and introduce a package for the R statistical software (R Development Core Team, 2011), while leaving the description of the mathematical details to others (Townsend, 1972, 1974; Townsend & Ashby, 1983; Townsend & Nozawa, 1995; Townsend & Wenger, 2004; Townsend & Honey, 2007; Dzhafarov, Schweickert, & Sung, 2004; Houpt & Townsend, 2010b, 2011, 2012). We begin by outlining the general purpose of SFT and the specific questions the methodology can and cannot address. We then describe the measures in SFT, the Mean and Survivor Interaction Contrasts and the workload capacity coefficients. We cover the necessary experimental manipulations to use the measures, the statistical tests associated with the measures, and how to use the R package for the measures. We will end with a description of the Double Factorial Paradigm, an experimental setup that allows one to use both the Survivor Interaction Contrast and the capacity coefficients. In each section, we will describe the relevant functions in the **sft** R package and give a brief example. For reference, a complete list of the functions currently implemented in the package is given in Table 1.

The Goals of SFT

As stated in the introduction, SFT is a framework for understanding the cognitive processing of multiple sources of information. These multiple sources could take the form of information from different modalities, such as audio and visual information, or different properties of a stimulus within

Joseph W. Houpt
Department of Psychology
Wright State University
3640 Colonel Glenn Highway
Dayton, Ohio 45435
joseph.houpt@wright.edu

This work was supported by AFOSR grant 10RH07COR to the late D. W. Repperger and P. R. Havig. We would like to thank Chris Myers for his comments on an early version of this paper. **Distribution A.** Appproved for public release; distribution unlimited. Approval given by WPAFB Public Affairs. Disposition Date: 19 November 2012, Document Number 88ABW-2012-6052.

Function	Description
capacity.and	Calculates the capacity coefficient for exhaustive (AND) processing.
capacity.or	Calculates the capacity coefficient for first-terminating (OR) processing.
capacityGroup	Performs workload capacity analysis on each participant and each condition.
estimateNAH	Calculates the Nelson-Aalen estimator of the cumulative hazard function.
estimateNAK	Calculates the Nelson-Aalen estimator of the cumulative reverse hazard function.
estimateUCIPand	Estimates the cumulative reverse hazard function of an UCIP process on an AND task.
estimateUCIPor	Estimates the cumulative hazard function of an UCIP process on an OR task.
mic.test	Performs either an Adjusted Rank Transform or ANOVA test for an interaction at the mean level.
sic	Calculates the Survivor Interaction Contrast and associated measures.
sic.test	A nonparametric test of for significant positive and negative parts of an SIC.
sicGroup	Performs SIC analysis on each individual and each condition of a DFP experiment.
siDominance	Tests for the ordering of survivor functions implied by selective influence using KS tests.
ucip.test	A nonparametric test for capacity values significantly different than those predicted by the estimated UCIP model.

Table 1
A complete list of the functions included in the **sft** *package with a brief description. The capacity and sic functions are explicated in the text below. UCIP refers to unlimited capacity, independent, parallel. For more details on the syntax for each function, the manual is available on the Comprehensive R Archive Network (http://cran.r-project.org/web/packages/sft/sft.pdf).*

a modality, such as color and shape. In many cases the sources are based on the experimental design and not necessarily on psychologically meaningful features, such as the top half and bottom half of a face (cf. Burns, Pei, Houpt, & Townsend, 2009). Questions about how the sources are processed together can be grouped into four classes: architecture, stopping rule, stochastic dependence and workload capacity, each of which will be defined below (cf. Townsend, 1974).

Before going into the details of each of those properties, we first want to point out that SFT is not designed for studying a single source in isolation. Psychologists are often interested in how changes in a single dimension of a stimulus correspond to changes in performance. This includes finding psychometric curves, just noticeable differences, etc. SFT is not the appropriate tool for these questions. The SFT approach is focused on multiple sources of information, each of which can be used to make a response. Additionally, the theoretical tools described in this paper were developed for high accuracy tasks, although the general theory can be extended to include variations in accuracy (e.g., Townsend & Altieri, 2012).

Architecture

Within the SFT framework, architecture refers to the temporal organization of the processes.[1] Within this approach we are interested in assessing if the processes fall into one of a few broad classes of architectures, in order to qualitatively characterize the system structure. One such class, foundational to the early work in response time research (e.g., Donders, 1969), is a serial architecture. When a serial architecture is employed, each source of information is processed one at a time in a sequence, i.e., serially. For example, if a participant is watching the left and right side of a display to

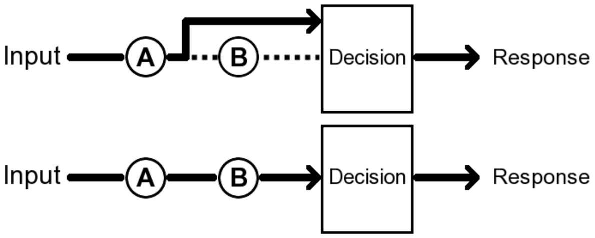

Figure 1. Serial architectures with OR (above) and AND (below) stopping rules. In serial processes, each target is processed sequentially. In an OR process, the participant can stop once either A or B has completed. In an AND process, both A and B must complete before a response is made.

[1] In other areas of cognitive modeling, architecture is used to refer to fixed properties of the cognitive system. In some cases, this may include the temporal organization of the information processing, but they are distinct concepts. Architecture in the sense of this paper may vary with a participant's strategy, especially on high level cognitive tasks in which a participant has a fair amount of control over strategy (Fifić, Nosofsky, & Townsend, 2008; C.-T. Yang, 2011; C.-T. Yang, Hsu, Huang, & Yeh, 2011; C.-T. Yang, Chang, & Wu, 2012). Architecture in the other sense could refer to properties that we classify under other monikers, such as workload constraints on information processing efficiency.

determine if a small dot appears (see Townsend & Nozawa, 1995), a serial architecture would mean she first checks if the dot has appeared on one side, then the other. Note that a serial architecture does not necessarily mean the dots are checked in a specific order or even the same order each time, but only that one side is checked before the other on each trial. Two types of serial architecture are depicted in Figure 1. The difference between these serial processes is the stopping rule, which is discussed in the next section.

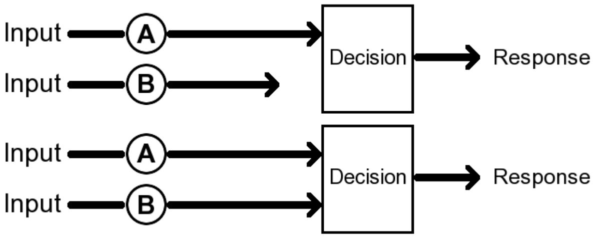

Figure 2. Parallel architectures with OR (above) and AND (below) stopping rules. In parallel processes, each target is processed at the same time. In an OR process, the participant can stop once either A or B has completed. In an AND process, both A and B must complete before a response is made.

In contrast, both sources of information may be processed simultaneously, i.e., in parallel. With a parallel architecture, the participant watching the monitor would concurrently check both the left and right sides for the appearance of a dot. Parallel architectures are depicted in Figure 2. Like the serial models in Figure 1, the difference between these parallel processes is the stopping rule, which is discussed in the next section.

As another example of the distinction between parallel and serial processing, consider a categorization task in which categories are determined by two different dimensions, say color and form (see Fifić, Little, & Nosofsky, 2010). In a serial process one would first check which category is indicated by the shape (color) of the object, then check which category is indicated by the color (shape). In a parallel process, both color and shape are examined at the same time.

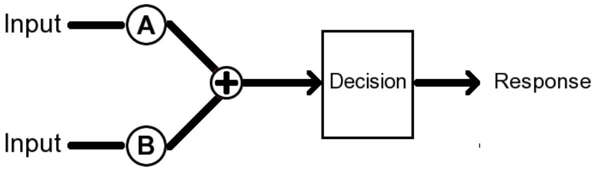

Figure 3. A coactive model, based on the summed activation of the channels.

A special case of parallel processing, referred to as coactive processing, is when the information is pooled before a decision is made. In a standard parallel model (e.g., those depicted in Figure 2), a decision about each source of information is made separately; then each of those individual decisions are combined (usually according to a stopping rule, defined below) for the final decision or action. In a coactive architecture, all of the information is used together to directly make the final decision. The pooling can take different forms, although the most well developed coactive models assume the information is summed and the total is compared to a threshold to make a decision (e.g., Schwarz, 1989, 1994; Townsend & Nozawa, 1995; Houpt & Townsend, 2011). This coactive model is depicted in Figure 3.

An interesting example of a case when coactive processing is possible is when information sources from different modalities are processed together (e.g., Miller, 1982). Suppose you need to determine if someone speaking had said “bad” or “dad” based on seeing the speaker’s mouth and hearing the word spoken (see Altieri & Townsend, 2011). If you first check for the difference between the “b” and “d” based only on what you hear, then based only on how the speaker’s mouth looked when she pronounced the word, this would be a serial process. If you simultaneously process the visual cues and the aural cues, you are using a parallel process. Coactive processing is the special case in which, instead of separately determining if the mouth indicates “bad” and the sound indicates “bad,” you pool evidence from both modalities to make the decision.

Other architectures may be possible, particularly with more than two sources of information. Many of those possible architectures can be formulated in terms of combinations of parallel and serial processes. These more advanced models are beyond the scope of this paper, but the theoretical work on SFT for more than two sources continues to evolve.

Stopping Rule

A second question that arises with respect to the combination of multiple sources of information is that of the stopping rule, or how many sources of information are processed before a person responds.[2] Similar to the SFT approach to architecture, the SFT approach is concerned with the qualitative classes of stopping rule, rather than exact quantitative measures of information, and so the methods will assess which class of stopping rule is engaged given a particular

[2] In many models, the amount of information required to stop processing a given source (i.e., the threshold in an information accumulator model, cf. Link & Heath, 1975; Ratcliff & Smith, 2004; Brown & Heathcote, 2008) can vary. This also falls under the general category of stopping rule. However, the SFT approach does not include the more detailed analyses involved in identifying changes in the amount of information needed for each sub-process to finish.

task. One possibility is that a person must exhaustively process all of the information available before responding. We will often refer to this as AND processing, in reference to the Boolean rule combining the decision on each source, but it is also known as an exhaustive or maximum time stopping rule. AND stopping rules can be used with serial models (bottom of Figure 1) and parallel models (bottom of Figure 2).

Alternatively, a person may respond as soon as he detects a target, regardless of how many other sources of information are present. This stopping rule is often referred to as self-terminating. Self-terminating stopping rules encompass a variety of possible decision situations ranging from decisions based on a single target among distractor information to needing to identify a subset of multiple targets for a decision, although not all targets as in the AND case. In the special case in which all sources of information indicate a target (also known as a redundant-targets task), a person can respond as soon as any one source is finished processing. This is referred to as a first-terminating process, or often, simply OR, again in reference to the Boolean logic rule. Self-terminating stopping rules can be combined with serial models (top of Figure 1) and parallel models (top of Figure 2).

The design and demands of an experiment will often require a participant to use a particular stopping rule to correctly respond. For example, if participants were asked to respond positively only if they detected *both* a dot above the midline and a dot below the midline of a display, then they would need to check both above *and* below before responding (e.g., Townsend & Nozawa, 1995; Eidels & Townsend, 2009). If the instructions change so that a positive response is made if there is a dot detected in *at least one* position, above *or* below (or both), then the task no long requires the AND stopping rule but allows for self-terminating responses. However, even when the task allows for a self-terminating process, people may still exhaustively process the information, as observed in some clinical populations (e.g., Johnson, Blaha, Houpt, & Townsend, 2010).

Stochastic Dependence and Selective Influence

Another well-studied property of combined information processing is the extent to which each process depends on the others. We formalize this construct with the probabilistic definition of independence. If the distribution of the processing times of all of the sources is equal to the product of the distributions of the processing time of each individual source, we say that they are independent. Otherwise, they are dependent.

A closely related, although not identical concept is that of selective influence. An experimental manipulation selectively influences the processing of a source if that factor changes the processing of that source, but processing of all other sources is unchanged. In the dot detection experiment described above, selective influence of the contrast manipulation of a dot would mean that reducing or increasing the contrast of the dot above fixation does not change how quickly dots below fixation are detected and vice versa. For a more formal treatment of selective influence, see Dzhafarov (2003) and Dzhafarov and Gluhovsky (2006).

There are different ways that the independence might fail, some of which may also cause failures of selective influence. One way that a dependence between the processing times could arise is when an external factor, such as attention, speeds up or slows down the processing of all of the sources. When a person is more focused on the task, processing times will be faster for all sources. When he is less focused, processing times will be slower. Thus, if the processing time for one source is known to be fast, it is more likely that the person was focused and hence processing times for the other sources are likely to be faster. Therefore, the processing times are dependent. Despite this dependence, selective influence may still hold, as long as the experimental manipulation does not affect the participant's attention.

Completion times may also be dependent due to interactions among the processes (e.g., Townsend & Wenger, 2004; Eidels, Houpt, Pei, Altieri, & Townsend, 2011), which will also lead to failures of selective influence (cf. Townsend & Thomas, 1994). For example, with configural stimuli, such as faces (Fifić & Townsend, 2010) or words (Houpt & Townsend, 2010a), different sources of information within the stimulus can facilitate each other. One way this may occur in faces is that the more detail one perceives from the left side of a face image, the more information one has about the right side of that same face image. Facilitation among sources of information can also occur when participants are highly trained with a stimulus such that the components are unitized and practiced (e.g., Blaha, 2010).

A further potential source of dependence arises if participants discern, even implicitly, conditional stimulus probabilities based on the presentation rates of different stimulus elements (Mordkoff & Yantis, 1991). This dependence can be mitigated by careful experimental design, and we will return to discuss this issue in greater detail in the 'Stimulus Rates and Contingencies' section.

Workload Capacity

A fourth property characterizing the processing of multiple sources is how the processing rate of each source changes as more sources are added. This characteristic is termed workload capacity, which is a special case of the general system information processing capacity (see Townsend & Ashby, 1983). Note that in the SFT context, 'capacity' refers to the information throughput characteristics of the system, addressing the question of how much work can be completed (i.e., information processed) in a given amount of time. Additionally, 'workload' refers to the manipulation of the number of sources of information (e.g., number of stimulus modalities, number of features in an object). Thus, workload capacity assesses how much information is processed over time when the amount of information available to be processed is manipulated. For example, a recent study applied the workload capacity construct to inform models of the Thatcher illusion (Donnelly, Cornes, & Menneer, 2012). They showed that, although participants were faster at detecting a manipulation of the features when multiple manipulations were present in the stimulus, there was no evidence that the processing of each feature had changed. This result is evidence against positive interactions between the feature-specific processes, a common explanation of the Thatcher illusion.

As with architecture and stopping rule, the SFT approach is to qualitatively assess any changes in processing rates by classifying capacity into one of three categories: limited, unlimited, and super capacity. Limited capacity processing is when performance on each individual source degrades as the number of sources increases. This degradation is typically interpreted as a slowing of individual information sources' processing rates in the presence of additional sources. Unlimited capacity refers to performance which reflects no effect of an increased workload on each individual process (i.e., the additional sources do not influence the processing rate of the original information sources). Finally, super capacity is when performance on each source is better under increased workloads, meaning that the addition of more information sources has resulted in increased processing speed for the other individual sources. Humans are not likely to be unlimited capacity, let alone super capacity, with a large number of sources. Nonetheless super capacity is often observed in experimental settings using a limited number of sources (e.g., Houpt & Townsend, 2010a; Blaha, 2010). In some cases, this is the result of facilitation (a positive stochastic dependence) among the processes. Another situation that would result in super capacity is if participants exhibit less focused attention on single source tasks, perhaps because they are relatively easy, but focus more attention on the task when there are multiple sources present.[3] Further research on how super capacity can arise is an exciting direction for SFT in applied cognitive psychology (cf. Repperger et al., 2009). Under this capacity taxonomy, the aforementioned Thatcher illusion study by Donnelly et al. (2012) found unlimited capacity, despite the popular notion that the perception of face parts is super capacity when the parts are presented in their normal configuration.

The Measures in SFT

Having delineated the basic processing aspects, we now turn to the SFT measures available for analyzing these properties. We note that all four of the constructs above are independent properties, so a system can be defined by any combination of architecture, stopping rule, (in)dependence, and workload capacity level. If it were possible to directly observe the time it takes for each process to complete, analysis would be greatly simplified. Unfortunately, observed response times are based on the aggregate of the various properties. For example, when an audio target and visual target are present in an OR task, a given response time could be produced by any combination of each of the properties above (Townsend, 1972; Townsend & Ashby, 1983). Even if we know how long responses to audio and visual targets take in isolation, all combinations are still possible, although some trade-offs are necessary depending on the observed response times on the OR task. For example, slower response times in the redundant target condition could be due to limited-capacity, parallel processing or unlimited-capacity, serial processing. It is therefore critical to find ways to analyze all four properties simultaneously from a single set of measurements.

SFT includes measurements that are informative with regards to architecture, stopping-rule, workload capacity and stochastic dependence. The Mean Interaction Contrast and the Survivor Interaction Contrast are tools for analyzing architecture and stopping rules and can, in some cases, also be informative about stochastic dependencies. The capacity coefficients for OR processes, $C_{\text{OR}}(t)$, and AND processes, $C_{\text{AND}}(t)$, are useful for measuring workload capacity and stochastic dependencies; some inferences about architecture may also be possible from the capacity coefficients.[4]

One important feature of the SFT measures is that they are nonparametric in nature, thereby enabling researchers to investigate the information processing properties for a given task without any parametric assumptions about the response time distribution. All of the measures present here are based on some transformation of the empirical response time distribution.

The Survivor Interaction Contrast and Mean Interaction Contrast

The Survivor Interaction Contrast (SIC; Townsend & Nozawa, 1995) indicates the architecture and stopping-rule of the underlying information processing system. To estimate the SIC for a participant in a given task, response times are needed from conditions in which the speed of processing each individual source of target information is factorially manipulated. Then an interaction contrast of the estimated survivor functions of the response times for those conditions is taken.

[3] We do not wish to claim that participants frequently, or even ever, attend more to the task when there are multiple sources of information. We only wish to indicate that it is not entirely an unreasonable possibility.

[4] Each of the tools in isolation are relatively weak with respect to analyzing stochastic dependence, but are powerful when used together (Eidels et al., 2011).

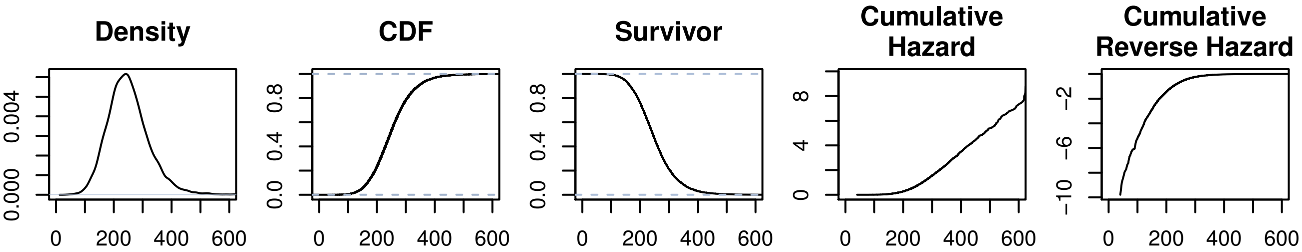

Figure 4. Different functions describing the same random variable (in this case, an exGausian random variable). On the far left is the probability density function (or PDF). Next is the CDF, $\Pr\{X \leq t\}$, then the survivor function, $\Pr\{X > t\}$. The final two graphs are the cumulative hazard function, $H(t)$, and the cumulative reverse hazard function, $K(t)$.

The survivor function, $S(t)$, is the probability that an event has not yet occurred by time t, i.e., the survivor function of a random variable X is $S_X(t) = \Pr\{X > t\}$. See Figure 4 for a depiction of the various descriptions of a random variable. For response times, it is the probability that a participant has not responded by a given time. $S(t)$ is the complement of the more familiar cumulative distribution function (CDF), $F_X(t) = \Pr\{X \leq t\}$,

$$F_X = \Pr\{X \leq t\} = 1 - \Pr\{X > t\} = 1 - S_X(t).$$

Much of the early work with the SIC focused on simple visual detection tasks, so an experimental manipulation to speed up and slow down processing is frequently referred to as a salience manipulation. Conditions that should lead to faster processing are usually denoted by an "H" for high salience; slow conditions are usually denoted by an "L" for low salience. For example, $\mathrm{S}_{\mathrm{HL}}(t)$ is the survivor function of the response times when the first target is high salience and the second target is low salience.

With all of the notation in place, we now can state the SIC for two sources of target information,

$$\mathrm{SIC}(t) = [\mathrm{S}_{\mathrm{LL}}(t) - \mathrm{S}_{\mathrm{LH}}(t)] - [\mathrm{S}_{\mathrm{HL}}(t) - \mathrm{S}_{\mathrm{HH}}(t)]\,. \qquad (1)$$

Each of the two parts of the contrast in brackets should generally be positive: Response times in a low salience condition should tend to be slower than in a high salience condition, and slower response times lead to larger survivor functions relative to the high salience conditions. While this ordering does not always hold, the assumption of effective selective influence is sufficient to guarantee the survivor functions are ordered as expected, $\mathrm{S}_{\mathrm{HH}} < \{\mathrm{S}_{\mathrm{LH}}, \mathrm{S}_{\mathrm{HL}}\}$ and $\mathrm{S}_{\mathrm{LL}} > \{\mathrm{S}_{\mathrm{LH}}, \mathrm{S}_{\mathrm{HL}}\}$ (see Figure 5).[5] Effective selective influence does not imply any particular relationship between S_{HL} and S_{LH} and their order does no effect any conclusions about the models.

When effective selective influence holds, each of the parallel and serial models with OR and AND stopping rules, have unique SIC forms regardless of the distributions of the individual channel completion times (Townsend & Nozawa, 1995; Dzhafarov et al., 2004). Figure 6 depicts these SIC forms. A parallel model with an OR stopping rule has an entirely positive SIC. A parallel model with an AND stopping rule has the opposite, an entirely negative SIC. Serial processes with OR stopping rules have flat SICs, equal to zero for all times. A serial process with an AND stopping rule is first negative, then positive, producing an s-shaped signature.

Additionally, two types of coactive models, one based on Poisson processes (Townsend & Nozawa, 1995) and the other based on diffusion processes (Houpt & Townsend, 2011), also have SIC forms that differ from the serial and parallel models. Like the serial-AND models, these coactive processes have an SIC that is first negative then positive. The SIC for a coactive model is shown in Figure 7. The feature that distinguishes the serial-AND and coactive models is the relative negative and positive areas under the SIC curve; serial-AND processes have equal positive and negative areas while coactive processes have more positive area under the SIC curve. To measure the area under the curve, we use the integrated SIC. Due to a useful property of positive random variables (such as response times), the integrated SIC turns out to be an interaction contrast of the mean response times,

$$\mathrm{MIC}(t) = [\mathrm{M}_{\mathrm{LL}}(t) - \mathrm{M}_{\mathrm{LH}}(t)] - [\mathrm{M}_{\mathrm{HL}}(t) - \mathrm{M}_{\mathrm{HH}}(t)]\,. \qquad (2)$$

Thus, when the SIC exhibits an s-shape, a positive Mean Interaction Contrast (MIC) indicates a coactive process and a zero MIC indicates a serial process with an AND stopping rule.

Other SIC forms can also arise from channel interactions (Eidels et al., 2011). These interactions lead to violations of selective influence, so the SIC forms are no longer required to be those shown in Figure 6. Instead, depending on the

[5] The additional adjective "effective" simply means that the salience manipulation has an effect: Channel processing times should be faster when the input is high salience. Here we mean a particularly strong type of faster, that the response times for the fast condition should stochastically dominate the response times for the slow condition: $S_H(t) \leq S_L(t)$ for all t with $S_H(t) < S_L(t)$ for at least some t.

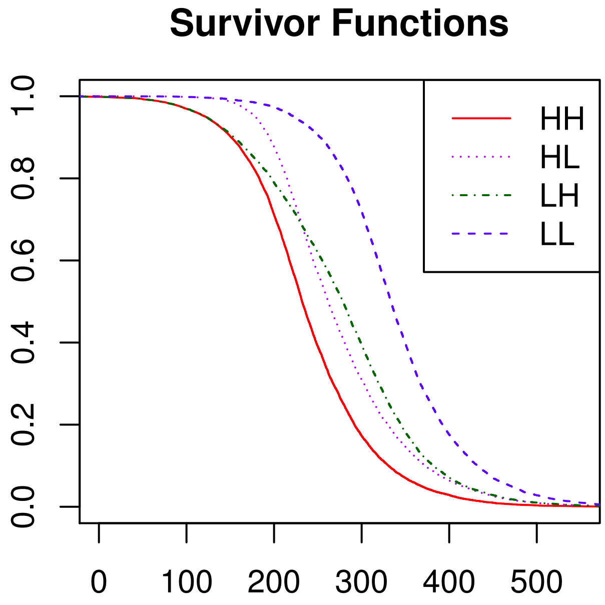

Figure 5. An example of survivor functions with the order implied by effective selective influence. HH indicates that both targets are processed relatively fast, i.e., are high salience. HL indicates that one target is high salience and the other is low salience; LH and LL follow the same scheme. Note that the relationships $S_{HH} < S_{HL}$, $S_{HH} < S_{LH}$, $S_{LL} > S_{HL}$, and $S_{LL} > S_{LH}$ are implied by selective influence, but a specific ordering relationship between S_{HL} and S_{LH} is not implied.

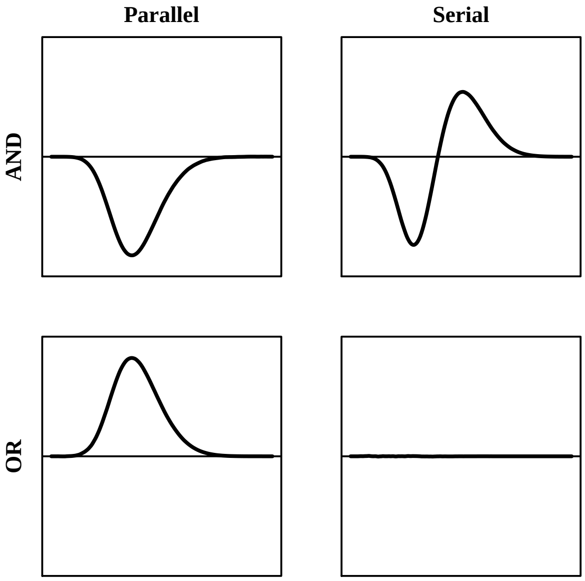

Figure 6. Survivor Interaction Contrast predictions for parallel and serial models with AND and OR stopping rules, assuming selective influence.

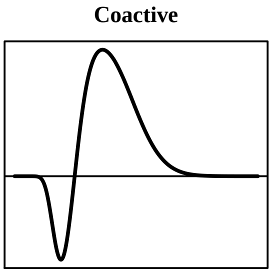

Figure 7. Survivor Interaction Contrast prediction for the Poisson and diffusion based information summing coactive models.

degree of interaction, parallel-facilitatory models can have SIC shapes ranging from the predicted independent form to matching the s-shaped coactive form. Inhibitory, parallel models with OR stopping rules remain entirely positive, whereas inhibitory, parallel models with AND stopping rules can have SIC forms ranging from entirely negative to nearly entirely positive. The analogous work on interactive serial models is yet to be completed.

SIC in R. The **sft** package in R (R Development Core Team, 2011) includes the function **sic** to calculate the SIC and MIC.[6] The function takes response times from each of the salience conditions, HH, LH, HL and LL, and returns a **stepfun** object representing the estimated SIC.[7] The function returns additional useful information for interpreting the SIC. It includes the results of a series of Kolmogorov-Smirnov tests for distribution ordering used to check for the expected ordering of survivor functions mentioned above, $S_{HH} < \{S_{LH}, S_{HL}\}$ and $S_{LL} > \{S_{LH}, S_{HL}\}$. The survivor ordering test can also be run in isolation, using the function **siDominance**.

The **sic** function also performs a statistical analysis to determine whether the positive and negative parts of the SIC are significantly different from zero. Currently the only statistical test of the SIC is based on the generalization of the two-sample Kolmogorov-Smirnov test, as described in Houpt and Townsend (2010b).[8] This test performs two null-

[6] We will cover the **sft** functions together with the relevant theory and definitions without detail regarding data formats; we address the formatting of data for use in **sft** in a later section.

[7] The calculation of the SIC is based on the R function **ecdf**. Both the **ecdf** function and the **stepfun** class are included in the stats package as part of R (R Development Core Team, 2011). For details on these, or any other function or class in R, we suggest the use of the **help** function.

[8] Some researchers have attempted to apply bootstrapping for

hypothesis tests to separately assess the positive-going deviations from $SIC(t) = 0$ and the negative-going deviations from $SIC(t) = 0$. The first test for whether the largest positive value of the SIC (D^+) is significantly different from zero is a one-sided test with $H_0 : D^+ \leq 0$ and $H_a : D^+ > 0$; the result of this test is reported as **sic$ positive** with both the test statistic value and exact *p*-value returned. The second test is a one-sided test for whether the largest negative value (D^-) is significantly different from zero, with $H_0 : D^- \geq 0$ and $H_a : D^- < 0$. The result of this test is reported as **sic$ negative** with both the test statistic value and exact *p*-value reported.

There is also a separate function, **mic.test**, for performing a two-tailed test of the MIC value under the null hypothesis that $MIC = 0$ against the alternative hypothesis that $MIC \neq 0$. There are two options, if **ART=TRUE** the adjusted rank transform test (ART; Sawilowsky, 1990; Reinach, 1960), a nonparametric test, is used. If **ART=FALSE**, then an ANOVA is used. Like the **sic** function, **mic.test** takes response times from each of the salience conditions as input. It then returns the exact *p*-value and test statistic from the chosen test.

Example 1 demonstrates the use of the **sic** function applied to data simulated from a Serial-AND model. The survivor dominance tests all indicate proper ordering, meaning the SIC shape should be interpretable. Both D+ and D- are significant, indicating that the SIC has both positive and negative parts. Both Coactive and Serial-AND models predict significant positive and negative parts, so we also check MIC, which is not significantly different from zero. These results would lead us to reject parallel processes and Serial-OR models in favor of a Serial-AND model.

Example 1: **sic**

```
# Simulate single channel response times
> T1.h <- rweibull(300, shape=2, scale=400)
> T1.l <- rweibull(300, shape=2, scale=800)
> T2.h <- rweibull(300, shape=2, scale=400)
> T2.l <- rweibull(300, shape=2, scale=800)

# Combine into "observed" response times
# assuming a Serial-AND model
> hh <- T1.h + T2.h
> hl <- T1.h + T2.l
> lh <- T1.l + T2.h
> ll <- T1.l + T2.l

# Run the SIC analysis
> SerialAND <- sic(hh,hl,lh,ll)
> SerialAND
$SIC
Step function
Call: stepfun(RTall, c(0, sicall))
 x[1:1200] = 87.674, 89.236, 171.39,  ...,
1201 plateau levels = 0, 0.0033333, 0,

$Dominance
         Test   statistic      p.value
1 S.hh > S.hl 0.4100000 1.254669e-22
2 S.hh > S.lh 0.3866667 3.314705e-20
3 S.hl > S.ll 0.3300000 6.480362e-15
4 S.lh > S.ll 0.2800000 6.100898e-11
5 S.hh < S.hl 0.0000000 1.000000e+00
6 S.hh < S.lh 0.0000000 1.000000e+00
7 S.hl < S.ll 0.0000000 1.000000e+00
8 S.lh < S.ll 0.0000000 1.000000e+00

$positive
  Houpt-Townsend KS-SIC test
data:
HH: HH  HL: HL
LH: LH  LL: LL
D^+ = 0.17, p-value = 0.0131
alternative hypothesis: the SIC is above 0
 at some time

$negative
  Houpt-Townsend KS-SIC test
data:
HH: HH  HL: HL
LH: LH  LL: LL
D^- = 0.24, p-value = 0.0001769
alternative hypothesis: the SIC is below 0
 at some time

$MIC
  Adjusted Rank Transform test of the MIC
data:
```

hypothesis testing with the SIC, however there are problems with that approach. One can estimate *pointwise* confidence intervals, then check the confidence interval at each point to see if it includes zero. If one were to conclude that the function is significantly non-zero then the type I error rate will be much higher without appropriate correction. With a large number of estimated points on the SIC, a correction based on the assumption that each test is independent (e.g., Bonferroni) would make it nearly impossible to find a significant value of the SIC. Determining the appropriate correction based on the true dependencies among the points is possible, but it is more straightforward to simply treat the SIC as a function for hypothesis testing. Bootstrapping tests are possible for hypothesis about the function, but asymptotic tests (such as the Houpt & Townsend (2010b) test) are usually (always?) more powerful. Based on these issues, we have decided not to include bootstrap tests for SIC and C(t) measures in either the package or this paper.

```
HH: HH  HL: HL
LH: LH  LL: LL
MIC = 0, p-value = 0.8744
alternative hypothesis: the MIC is not zero

> plot(SerialAND$SIC, do.p=FALSE)
```

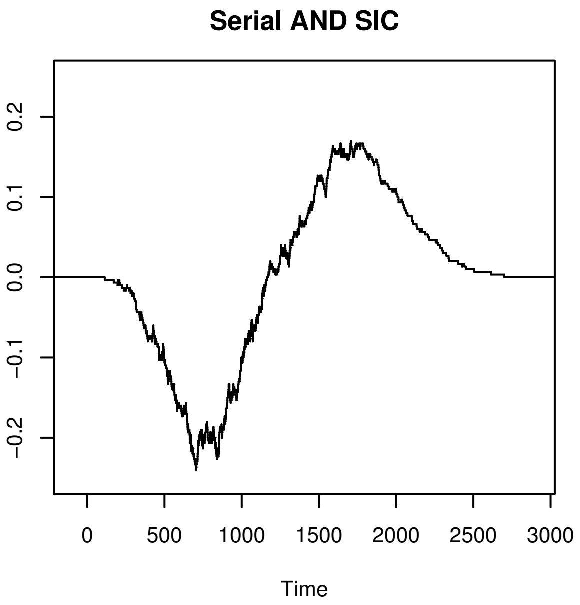

Figure 8. Plot of a simulated serial-AND SIC from running the **sic** function in Example 1.

The Capacity Coefficient

The capacity coefficients are based on the comparison of performance with multiple sources to a baseline calculated from performance with each single source of information. These functions can indicate variations in workload capacity, as well as dependencies among source processing times. Although a capacity coefficient could be defined for any stopping rule, the most commonly used are the OR capacity coefficient (Townsend & Nozawa, 1995) and the AND capacity coefficient (Townsend & Wenger, 2004). The baseline for comparison is based on the assumptions that processing of multiple sources is unlimited-capacity, independent and parallel (UCIP).

OR Processes. In an OR process, the probability that a response has not yet been made (the survivor function of the response times) is the probability that a target has not yet been detected on any channel. If we write $S_{AB}(t)$ for the survivor function of response times when both A and B are present targets and $S_{A(B)}(t)$ for the survivor functions of the channel completion times on A when the B target is present (and likewise for B in the presence of A), then, assuming the sources are independent,

$$S_{AB}(t) = S_{A(B)}(t) \times S_{B(A)}(t).$$

With the additional UCIP assumptions, the completion time distribution of A is unchanged regardless of whether B is present or not, $S_{A(B)}(t) = S_A(t)$ and likewise for B, $S_{B(A)}(t) = S_B(t)$. This is a situation commonly termed context invariance or context independence of the response time distributions. It follows from context invariance that a UCIP model predicts that the survivor function when both targets are present is equal to the product of the survivor functions for each target in isolation,

$$S_{AB}(t) = S_A(t) \times S_B(t).$$

The argument holds more generally; under the UCIP assumption the survivor function for any number of targets is the product of the survivor function for each of those targets in isolation,

$$S_{1\ldots n}(t) = \prod_{i=1}^{n} S_i(t).$$

For both statistical reasons (cf. Houpt & Townsend, 2012) and interpretability (cf. Townsend & Nozawa, 1995; Townsend & Ashby, 1983; Townsend & Eidels, 2011), the OR capacity coefficient is defined using cumulative hazard functions, $H(t)$.[9] To get from survivor functions to cumulative hazard functions one simply needs to take the natural logarithm, $\log[S(t)] = -H(t)$. Thus, because $\log(xy) = \log(x) + \log(y)$, the cumulative hazard function for the UCIP processing of n sources is

$$H_{1\ldots n}(t) = -\log\left[S_{1\ldots n}(t)\right] = -\log\left[\prod_{i=1}^{n} S_i(t)\right] = \sum_{i=1}^{n} H_i(t).$$

The OR capacity coefficient is defined as a ratio of a participant's actual performance when all sources are present, $\hat{H}_{1\ldots n}(t)$, to performance predicted from a UCIP system,

$$\mathrm{C_{OR}}(t) = \frac{\hat{H}_{1\ldots n}(t)}{\sum_{i=1}^{n} \hat{H}_i(t)}. \tag{3}$$

The denominator is the estimated cumulative hazard function for the UCIP model, derived from the response times for each process in isolation, and the numerator is the actual, observed performance with n target sources.[10]

[9] For details on the hazard function and its use in cognitive psychology, see Chechile (2003).

[10] We have not accounted for the additional time taken by non-perceptual, non-decision related processes, such as motor move-

Based on Equation 3, when the observed performance on all sources is equal to the UCIP model prediction, then $C_{\mathrm{OR}}(t) = 1$, and we can interpret the throughput for the system as exhibiting unlimited workload capacity. $C_{\mathrm{OR}}(t) < 1$ implies worse performance than the UCIP model, and we interpret this performance as the system exhibiting limited workload capacity. This indicates that either there are limited processing resources, there is inhibition among the subprocesses, or the items are not processed in parallel (e.g., the items may be processed serially). $C_{\mathrm{OR}}(t) > 1$ implies better performance than the UCIP model, in which case we interpret the throughput as exhibiting super workload capacity. This indicates that either there are more processing resources available per process when there are more sources of information, that there is facilitation among the subprocesses, or the items are not processed in parallel (e.g., the items may be processed coactively).

An alternative measure of capacity is based on the difference of cumulative hazard functions,

$$C_{\mathrm{OR}}(t) = \hat{H}_{1\ldots n}(t) - \sum_{i=1}^{n} \hat{H}_i(t). \qquad (4)$$

Although this form is non-standard, the variance function of the estimator can easily be calculated, unlike in the standard ratio form, thereby enabling a direct statistical test of $C_{\mathrm{OR}}(t)$ (see Houpt & Townsend, 2012 for details). The qualitative workload capacity interpretations of Equation 4 are the same as Equation 3, but the reference value is now 0 rather than 1 (e.g., $C_{\mathrm{OR}}(t) = 0$ is unlimited capacity, etc.).

OR Capacity in R. The Nelson-Aalen estimator of the cumulative hazard function (Aalen, Borgan, & Gjessing, 2008) can be calculated using the **estimateNAH** function in the **sft** package. It takes the response times as input, and, if desired, will also take an array indicating whether the participant was correct on each trial to adjust the estimate for incorrect responses. The function **estimateUCIPor** returns an estimate of a participant's cumulative hazard function when all targets are present, assuming UCIP processing, based on performance on each of the single target conditions. It takes a list as input in which each element is an array of the response times for each of the single target conditions. An additional list with the correct indicators for each condition can also be included. If the correct indicators are included, the estimates will be adjusted to account for incorrect responses (see Houpt & Townsend, 2012 for details). For both **estimateNAH** and **estimateUCIPor**, if the correct indicators are not provided, the function assumes all of the response times correspond to correct trials. Both **estimateNAH** and **estimateUCIPor** return estimated cumulative hazard functions, however **estimateUCIPor** returns an estimate of the cumulative hazard function if each sub-process were estimated using **estimateNAH** and the sub-processes are combined according to a UCIP-OR model.

The OR capacity coefficient and related statistical test (Houpt & Townsend, 2012) can be calculated using the **capacity.or** function. It takes as input a list containing arrays of response times from each condition (first the condition with all target sources present, then each of the single target source conditions) along with an optional list of correct indicators to use with the **estimateNAH** and **estimateUCIPor** functions. The function also includes an indicator, **ratio**, for whether to return the standard OR capacity coefficient (Equation 3) or the difference variant of the capacity coefficient (Equation 4).

The **capacity.or** function returns an **approxfun** object representing the standard ratio OR capacity coefficient ratio function (optional argument ratio=TRUE, which is the default) or the difference variant (ratio=FALSE) and the outcome of the **ucip.test** for OR processing. The **ucip.test** function returns the statistic value (a z-score) and p-value from a two-tailed test of the null-hypothesis of UCIP performance from Houpt and Townsend (2012). Note that if ratio=FALSE, **capacity.or** also reports the variance of the difference variant. If the reported p-value is less than your chosen Type I error α level, e.g., 0.05, then at least one of the UCIP assumptions has failed.

Example 2: **capacity.or**

```
# Generate single source response times
> rate1 <- 1/800
> rate2 <- 1/600
> pa <- rweibull(100, shape=2,
    scale=1/rate1)
> ap <- rweibull(100, shape=2,
    scale=1/rate2)

# Simulate a limited capacity
# Parallel-OR model
# Limited capacity means slower
# processing when there are multiple
# sources, so we use .5 times the
# original rate.
> pp.1 <- rweibull(100, shape=2,
    scale=1/(.5*rate1))
> pp.2 <- rweibull(100, shape=2,
    scale=1/(.5*rate2))
```

ments, in this derivation. This additional time would complicate the derivation, but it only has a limited effect on the capacity coefficient predictions when the variance of the additional time contributes relatively little to the variance of the response time which is reasonable for human data (Townsend & Honey, 2007). In particular, the extent to which the additional time changes capacity estimates scales with the variance of the base time, leading to underestimates of OR capacity (Townsend & Honey, 2007) and overestimates of AND capacity (Townsend & Eidels, 2011).

```
> pp <- pmin( pp.1, pp.2 )

#  Run the capacity analysis
> cap <- capacity.or(list(pp, pa, ap))
> cap$Ctest
Houpt-Townsend UCIP test

data:  RT and CR
z = -7.3626, p-value = 1.803e-13
alternative hypothesis: response times
are different than those predicted by
the UCIP-OR model

> plot(cap$Ct, xlim=c(0,2000), ylim=c(0,2))
```

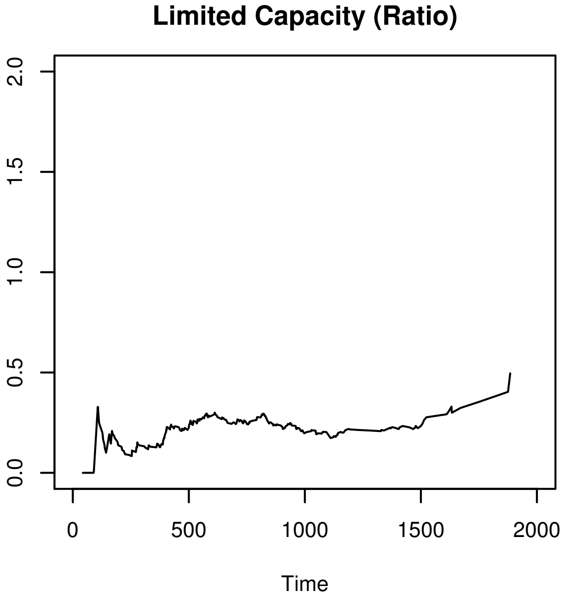

Figure 9. Plot of a simulated limited capacity coefficient from running the **capacity.or** function in Example 2.

AND Processes. In an AND process, the probability that a response was made (the CDF of the response times) is the probability that a target has been detected on all channels. If we write $F_{AB}(t)$ for the CDF of response times when both A and B are present targets, $F_{A(B)}(t)$ for the CDF of the channel completion times on A when the B target is present (and likewise for B in the presence of A), then, assuming the sources are independent,

$$F_{AB}(t) = F_{A(B)}(t) \times F_{B(A)}(t).$$

As part of the UCIP assumptions, the CDF of an individual target detection time is assumed to not change with respect to the presence of the other source, $F_{A(B)}(t) = F_A(t)$ and likewise for B, $F_{B(A)}(t) = F_B(t)$. Hence,

$$F_{AB}(t) = F_A(t) \times F_B(t).$$

More generally for n sources of information,

$$F_{1\dots n}(t) = \prod_{i=1}^{n} F_i(t).$$

Like the OR capacity coefficient, we take the natural logarithm of both sides to obtain capacity coefficient predictions. Because the AND model is in terms of the CDF rather than the survivor function, this results in a cumulative *reverse* hazard function, $\log[F(t)] = K(t)$.[11] Hence, the UCIP prediction for an AND task with n sources is,

$$K_{1\dots n}(t) = \log\left[F_{1\dots n}(t)\right] = \log\left[\prod_{i=1}^{n} F_i(t)\right] = \sum_{i=1}^{n} K_i(t).$$

The AND capacity coefficient is defined as a ratio of the participant's actual performance when all sources are present, $\hat{K}_{1\dots n}(t)$, to his predicted performance if he satisfied the UCIP assumptions,

$$\mathrm{C_{AND}}(t) = \frac{\sum_{i=1}^{n} \hat{K}_i(t)}{\hat{K}_{1\dots n}(t)}. \qquad (5)$$

The numerator is the estimated cumulative reverse hazard function for the UCIP model, derived from the response times for each process in isolation, and the denominator is the actual performance. The UCIP prediction is in the numerator for $\mathrm{C_{AND}}(t)$ so that the interpretation of values relative to one is consistent with $\mathrm{C_{OR}}(t)$. Note that this is because relatively larger cumulative hazard functions ($H(t)$) indicate faster processing, while relatively larger cumulative reverse hazard functions ($K(t)$) indicate slower processing.

As with Equation 3, unlimited capacity is exhibited by the system when $\mathrm{C_{AND}}(t) = 1$. $\mathrm{C_{AND}}(t) < 1$ implies worse performance than the UCIP model, interpreted as limited workload capacity. This indicates that either there are limited processing resources, there is inhibition among the subprocesses, or the items are not processed in parallel (e.g., the items may be processed serially). $\mathrm{C_{AND}}(t) > 1$ implies better performance than the UCIP model, interpreted as super workload capacity. This indicates that either there are more processing resources available per process when there are more processes, that there is facilitation among the subprocesses, or the items are not processed in parallel (e.g., the items may be processed coactively).

As with the OR capacity coefficient, there is also a difference variant,

$$\mathrm{C_{AND}}(t) = \hat{K}_{1\dots n}(t) - \sum_{i=1}^{n} \hat{K}_i(t). \qquad (6)$$

[11] For details on the reverse hazard function and its use in cognitive psychology, see Chechile (2011).

Again, this version allows the variance function of the estimator to be more easily calculated, enabling a direct statistical test of the null hypothesis $C_{\text{AND}}(t) = 0$ (Houpt & Townsend, 2012). The interpretation of Equation 6 remains the same as Equation 5, but the reference value is now 0 rather than 1 (e.g., $C_{\text{AND}}(t) = 0$ is unlimited capacity, etc.).

AND Capacity in R. The estimator of the cumulative reverse hazard function developed in Houpt and Townsend (2012) can be calculated using the **estimateNAK** function in the **sft** package. It takes the response times as input, and, if desired, will also take an array indicating whether the participant was correct on each trial to adjust the estimate for incorrect responses. The function **estimateUCIPand** returns an estimate of a participant's cumulative reverse hazard function for processing all targets, assuming UCIP processing, based on performance in each of the single target conditions. It takes a list as input in which each element is an array of the response times for each of the single target conditions. An additional list with the correct indicators for each condition can also be included. For both **estimateNAK** and **estimateUCIPand**, if the correct indicators are not provided, the function assumes all of the response times correspond to correct trials.

The AND capacity coefficient and related statistical test (Houpt & Townsend, 2012) can be calculated using the **capacity.and** function. It takes as input a list containing arrays of response times from each condition along with an optional list of correct indicators to use with the **estimateNAK** and **estimateUCIPand** functions. The function assumes that the first item in the list corresponds to the condition with all targets present and each subsequent item corresponds to a single target condition. The function also includes an indicator, **ratio**, for whether to return the standard AND capacity coefficient (Equation 5) or the difference variant of the capacity coefficient (Equation 6).

The **capacity.and** function returns an **approxfun** object representing the AND capacity coefficient ratio function (optional argument ratio=TRUE, which is the default) or the difference variant (ratio=FALSE) and the outcome of the **ucip.test** for AND processing. Note that if ratio=FALSE, **capacity.and** also reports the variance of the capacity coefficient difference variant. If the p-value is less than your predetermined Type I error α level, then at least one of the UCIP assumptions has failed.

Additional R Functionality. To run the statistical test without returning the additional **approxfun** objects for the capacity coefficient and variance, the function **ucip.test** is available. Like the capacity coefficient functions, it takes as input a list of response time arrays and a list of correct indicator arrays with the first element corresponding to the all targets present trials. The function also has a flag to indicate whether to test the data against UCIP OR processing (OR=TRUE) or UCIP AND process (OR=FALSE).

Example 3: **capacity.and**

```
# Generate single source response times
> rate1 <- 1/800
> rate2 <- 1/600
> pa <- rweibull(100, shape=2,
    scale=1/rate1)
> ap <- rweibull(100, shape=2,
    scale=1/rate2)

# Simulate an unlimited capacity
# Parallel-AND model
> pp.1 <- rweibull(100, shape=2,
    scale=1/rate1)
> pp.2 <- rweibull(100, shape=2,
    scale=1/rate2)
> pp <- pmax( pp.1, pp.2 )

#  Run the capacity analysis
> cap <- capacity.and(list(pp, pa, ap),
                        ratio=FALSE)
> cap$Ctest
Houpt-Townsend UCIP test

data:  RT and CR
z = -0.3614, p-value = 0.7178
alternative hypothesis: response times
are different than those predicted by
the UCIP-AND model

> plot(1:2000, cap$Ct(1:2000),
            ylim=c(-1,1), type='l')

# To run the statistical test, without
# returning the capacity coefficient
> ucip.test(list(pp,pa,ap), OR=FALSE)
```

There are also parametric methods for analyzing the capacity coefficient, based on fitting the Linear Ballistic Accumulator model (Brown & Heathcote, 2008) to the data. The procedures are outlined in Eidels, Donkin, Brown, and Heathcote (2010). We hope to add this functionality to the **sft** package in the near future.

The Joint Use of the SIC and $C(t)$

When the experimental manipulations selectively influence the intended process, the SIC gives a clear indication of the architecture and stopping rule of a system. When selective influence fails, such as would occur if there are interactions between the processes, it is more difficult to draw conclusions from the SIC alone. Eidels et al. (2011) demon-

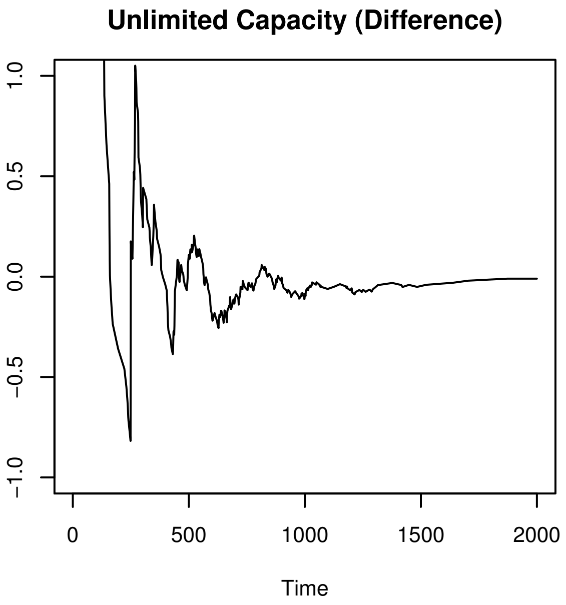

Figure 10. Plot of a simulated unlimited capacity coefficient from running the **capacity.and** function in Example 3.

strated that under some levels of interaction between processes, the SIC from one type of model can mimic the SIC signature of another model. For example, they simulated a Parallel-AND model with facilitation that produces a SIC that was indistinguishable from the serial-AND SIC (Figure 6, top-right).

While this mimicry may be problematic when attempting to use the SIC in isolation, Eidels et al. (2011) also show that the capacity coefficient can help to discriminate among model possibilities. Because the capacity coefficient compares performance to an unlimited capacity, independent, parallel baseline, facilitatory parallel systems will be relatively faster and thus exhibit super capacity relative to the UCIP baseline. In contrast, unlimited capacity, independent, serial processes (also referred to as the standard serial model) are relatively slower and exhibit have limited capacity coefficient values. Consequently, when a partially negative, partially positive SIC is observed and the MIC is zero, the capacity coefficient can be used to distinguish between facilitatory parallel and independent serial processes. For full details on the use of the SIC with the capacity coefficient, refer to Eidels et al. (2011), particularly Figures 3 and 4.

Although the SIC and capacity coefficient can be used together to analyze processing characteristics, it is important to remember that any combination of underlying characteristics is possible. For example, it is possible to have a limited capacity, facilitatory, parallel, exhaustive process. Whether the capacity coefficient will indicate super, unlimited, or limited capacity will depend on the relative degree of facilitation and workload capacity limitation. The form of the SIC will also depend on the degree of facilitation, ranging from all negative to nearly all positive. Nonetheless, the degree of facilitation does not in any way cause the architecture (or vice versa) just as the facilitation and architecture are distinct from the underlying workload capacity.

Clearly the interpretation of the data is more difficult when there are interactions among the processes, so whenever possible, it is best to use experimental factors that selectively influence the intended processes. When there is selective influence, the SIC forms in Figure 6 will hold regardless of the workload capacity level.

Designing Experiments for SFT

While all the SFT measures defined herein can be used individually, Townsend and Nozawa (1995) developed a single experimental paradigm integrating all the manipulations necessary to utilize both the SIC and $C(t)$ measures on a single set of data. This experimental design is known as the Double Factorial Paradigm (DFP). There are two critical types of manipulations that comprise the DFP: manipulation of workload and manipulation of salience. The name ‘Double Factorial’ is a reference to the use of a full factorial combination of the two manipulations with each manipulation incorporating at least two levels of each factor. In this section, we will illustrate two types of DFP designs for experiments with and without distractors in the stimuli. Example designs are shown in Figures 11, 12, and 13.

The first manipulation needed in the DFP is a workload manipulation. This manipulation is necessary to assess the workload capacity of the system, or the information throughput as the number of sources or targets increases. In a task with n possible sources of information (e.g,. $n = 4$ letters in a word as in Houpt & Townsend, 2010a), there are two critical levels of workload needed for the capacity coefficient: one condition in which all n sources/targets are presented simultaneously (e.g., all 4 letters presented as a word or letter string), and n conditions in which each of the sources/targets are presented individually (e.g., each single letter presented individually for a total of 4 single-source conditions). The latter condition is necessary for formulating the UCIP model predictions in the capacity coefficients (Equations 3 and 5), while the former is needed to compare actual performance to the UCIP prediction.

In its purest form, the workload manipulation is a change in the absolute number of physical items in the stimulus, such as the number of items in a visual search display or the number of features present in a face or object. However, there are times when is it not possible to have the pure absence of some stimulus characteristic, such as hierarchical forms wherein the global configuration would not exist without the local features, or when the experimental question demands the use of distractors in order to assess the information contained in the stimuli, like separable or integral dimensions. Workload in these cases can be considered a manipulation of the number of target information sources against either a neutral non-target or even distractor information. No matter the class of stimuli used, it is important that the aforementioned two condition types are present: a condition in which participants respond to all target sources together and all individual target conditions for estimating the UCIP model prediction.

The salience manipulation is needed to assess the processing architecture and stopping rule with the SIC (and MIC when needed). This manipulation changes the processing speed for the different targets or sources of information. The variation of the processing rates of each source of information enables inferences about the spatio-temporal arrangement of those subprocesses (i.e., the architecture) based on the measurement of a single response time. Often the choice of salience manipulation will depend on the task and the use of distractors in the class of stimuli. We will review below some example manipulations previously used in tasks with and without distractors. Regardless of the design, however, the choice of physical salience manipulation must meet the assumption of selective influence, i.e., each manipulation must affect only the speed of processing of one of the subprocesses of interest.

Example DFP without Distractors. Figure 11 illustrates a DFP experimental design for a simple visual detection task in which there are only targets and no possible distractor information from Eidels and Townsend (2009). This is a variation on the original DFP design by Townsend and Nozawa (1995) in which participants were asked to make responses about the detection of one or more dots of light presented in a dark environment. In this task, the workload manipulation is the presence or absence of two possible targets: a dot above fixation and a dot below fixation. When a target is absent, it is not replaced by a distractor, so the workload in this task, corresponding to the physical information in the stimulus, directly translates to the number of possible targets. With two possible targets, each of which could be present or absent, the whole task has three possible target workload levels: 0 (target absent), 1 (single target present) or 2 (double/redundant target present). The two single target conditions provide the data for the UCIP prediction in the capacity coefficient, and the double target present, also called redundant target, condition provides the data to be compared to the UCIP prediction in the capacity coefficients.

When a target is present in Figure 11, it can occur at one of two contrast levels, as illustrated by the lighter and darker grey dots, representing higher and lower contrast levels, respectively. This is the manipulation of target salience. The high contrast dots are well above detection threshold levels, appearing very bright to the participants, resulting in fast detection response times. The lower contrast level results in slower detection because it is not as bright as the high contrast level. Note that this low contrast does not need to be at the participant's absolute detection threshold, it only needs to be low enough to produce slower response times relative to the high contrast dots, in accordance with the selective influence assumption. In fact, it is desirable to find a lower contrast level here that will order the response time distributions according to selective influence but that still results in high detection accuracy.

When the workload manipulation is combined with the salience manipulation, there are multiple redundant target stimuli. In this dot task with two salience levels and two targets, there are four redundant target stimuli that provide the data for computing the SIC.

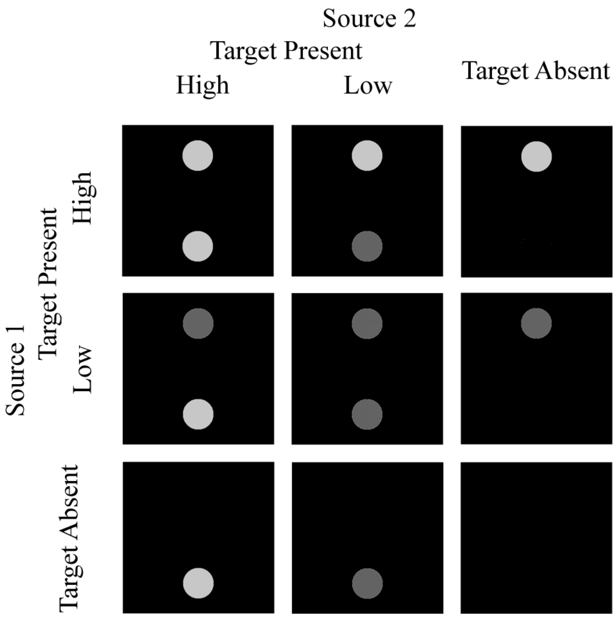

Figure 11. Double Factorial Paradigm simple visual detection task design. High salience of the dots is a higher contrast level against the black background than the low salience contrast level, which was closer to threshold detection levels. In this task, the light from the lower dot is one source and the light from the upper dot is the second source.

Note that the stimuli in Figure 11 can be used for both AND and OR decision rules, depending on the predetermined stimulus-response assignment structure. If participants are asked to respond when they detect the presence of any dot, then the response rule is OR because they can correctly respond 'yes' when presented with the the top dot alone, the bottom dot alone, or both dots together. If participants are asked to respond 'yes' only when they detect two dots on the screen, then they are following an AND stopping rule; they can only respond 'yes' correctly when the top and bottom dots are presented (upper left quadrant of Figure 11 only), and must respond no when zero or one dot is presented. Note that the decision rule for this task is not dependent on the salience manipulation of the targets.

Example DFP with Distractors. The DFP can be modified for use on tasks involving distractors in lieu of the presence/absence manipulation. There are two ways in which distractor information can be conceptualized in the DFP framework: as an uninformative placeholder for the absence of a target (especially when true absence is not possible) or as another source of information about the appropriate response. The use of distractors is particularly critical for tasks in which a pure absence manipulation is not possible, such as a global-local object discrimination task. An example of a hierarchical forms DFP design used by Johnson et al. (2010) is illustrated in Figure 12; we note here that the use of distractor information is critical in this task because a global object cannot exist without the local objects. Figure 12 illustrates a design in which the distractor is simply a placeholder: a single distractor dash is used to signify the absence of a target and provides a baseline shape from which the salience manipulation (the 'pointy-ness' of the arrowhead) can be defined (high is more pointy (a more pronounced arrowhead), while low is less pronounced and more similar to the dash). The assumption here is that because the distractor only has a single value, it is providing minimal competing information during the task. Thus, the single target conditions (global right arrow with local dashes and global dash composed of local right arrows) provide the single target conditions for the UCIP prediction in $C(t)$, and the redundant arrow conditions (global right arrows composed of local right arrows) provide the all-targets condition for $C(t)$ as well as the four redundant target stimuli (HH, HL, LH, LL) for the SIC and MIC calculations. In Johnson et al. (2010), participants were asked to make an OR decision similar to the above dot detection task: respond 'yes' when you see any right-pointing arrows, global or local or both, and respond 'no' only when you see no right-pointing arrows (the dash composed of dashes). But just as in the dot detection task, this same set of stimuli could be utilized in an AND decision task, where 'yes' responses are only to the redundant global-local right-pointing arrows.

Distractors can also be conceptualized as a second source of information and can be manipulated (number and salience) just as the target sources are. Such manipulation is critical for experiments addressing questions about the nature of information in multidimensional stimuli, such as whether the dimensions are integral or separable (Garner & Felfoldy, 1970; Garner, 1974). Manipulations of distractors are needed for modeling classification or discrimination tasks, for example, and variable distractors play a key role in visual and memory search tasks. Figure 13 illustrates a way to use distractors as additional, possibly competing, information sources with the global-local arrows. In this design, rather than the target absent being a single neutral non-target dash, the non-target distractors are left-pointing arrows that can also vary in salience. The task here could be an OR decision (respond 'yes' if there are any right-pointing arrows, global, local or both) or AND (respond 'yes' only when both the global and local arrows are pointing right). Interestingly in this example, when there is an AND decision on the redundant right pointing arrows, there is a corresponding OR decision being made about the left arrows. When there is an OR decision on the right arrows, there is a corresponding AND decision on the redundant left arrows. Additionally, for assessing response biases and counterbalancing the experiment, the experimenter can leverage this to provide instructions emphasizing the AND or OR rule for the right or left arrows simply by changing the experimenter-made assignment of arrow direction to target/distractor class or response key. Because both the targets and distractors are manipulated in their number and salience, it is possible to model both types of stimuli with the SFT measures to enumerate the processing characteristics of both the targets and the distractors, although we cannot say whether or not the same mechanisms are engaged in both cases.

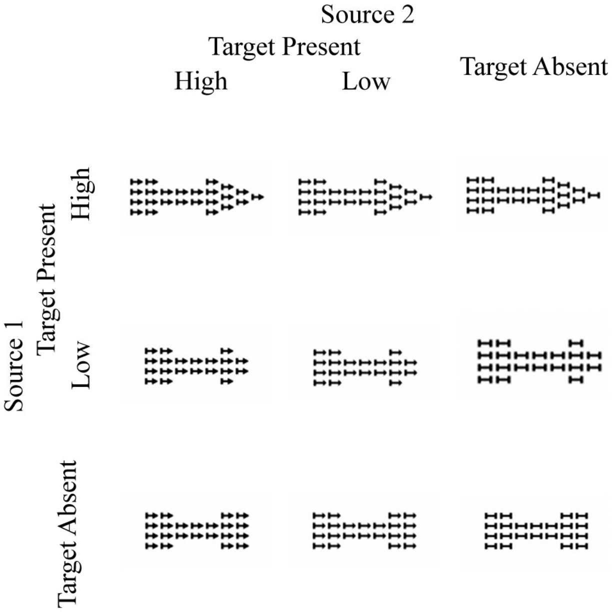

Figure 12. Global-local arrow stimuli in the target present/absent DFP design from Johnson, et al. (2010). Arrow salience is defined relative to the neutral dash that points neither left nor right. The low salience right pointing arrows have smaller arrowheads and are harder (slower RT) to distinguish from the dash than the high salience right pointing arrows with the more pronounced arrowheads.

Stimulus Rates and Contingencies. Consideration must be given early in the DFP design process to the presentation rates for each stimulus or stimulus type (redundant/double targets, single targets, target absent, etc.), in or-

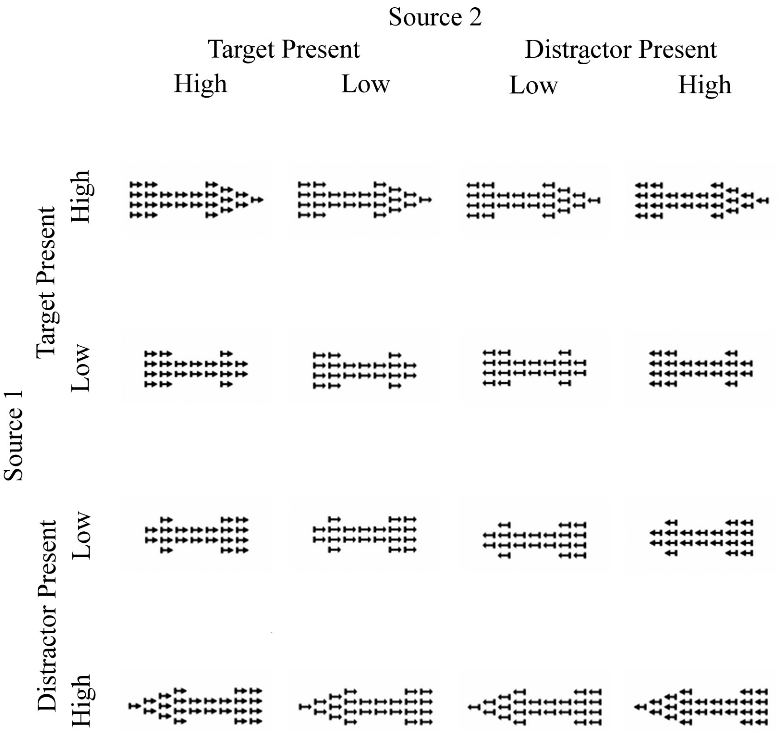

Figure 13. Global-local arrow stimuli in the target versus distractor discrimination DFP design where targets are right-pointing arrows and distractors are left-pointing arrows. Arrow salience is defined by the relative sizes of the arrowheads which make the direction harder (slower RT) or easier (faster RT) to determine.

der to minimize or eliminate two sources of bias that can arise from stimulus contingencies. It is possible that correlations between different target sources and/or between target and non-target items can be introduced through the rates of presentation for the target and non-target features/stimuli/sources. Participants can often, even implicitly, pick up on such correlations, which can then bias task performance. Of particular concern to the present effort is that, within the DFP, it is not possible to eliminate all non-zero contingencies, as we will illustrate below. Thus, as experimenters, we must be aware of these correlations, particularly if they provide any advantage to redundant target conditions, because such advantages have been shown to affect the size of the redundant-signals effect and influence the comparison between parallel and coactive models (Mordkoff & Yantis, 1991). As we discuss the possible contingencies, we will review the stimulus rates for DFP that will minimize the amount of rate-based contingencies present in target detection tasks and target/distractor discrimination tasks.

The statistical relationships between the various targets and non-targets are called inter-stimulus contingencies (ISCs), which can result in stochastic dependencies in the form of interchannel crosstalk between the information sources. Mordkoff and Yantis (1991) describe mathematical expressions that can be used to derive the ISCs that may be present in an experimental design. ISC computations are based on the difference between the conditional probability of one stimulus element given a second element and the base rate of the first stimulus element. In general, it is desirable for all potential ISCs in an experimental design to be 0, so that there is no potential for biasing performance based on the presence or absence of a particular stimulus element. That is, knowing that a target or non-target is present in one channel should give you no information about whether or not a target is present in the other channel. For a full discussion of the influence ISCs can have on race model performance and model identification, refer to Mordkoff and Yantis (1991).

For our discussion of ISCs and to foreshadow the coding scheme for the **sft** package, we will use the following notation:

A The first source of information to be processed.
B The second source of information to be processed.
A^2 High salience target stimulus on first source.
A^1 Low salience target stimulus on first source.
A^0 No stimulus on first source of information.
A^{-1} Distractor stimulus on first source of information.

Using this notation, the ISC between a non-target and a target element is given by

$$ISC(A^0 \Rightarrow B^1) = P[B^1|A^0] - P[B^1]$$

or

$$ISC(B^0 \Rightarrow A^1) = P[A^1|B^0] - P[A^1].$$

There are two ISCs of interest in the basic detection DFP illustrated in Figures 11 and 12: the relationship between a target in one channel and a non-target element in the other, and the relationship of the target elements to each other. The ISCs of interest are:

$$\begin{aligned}\text{ISC}(A^0 \Rightarrow B^1) &= P[B^1|A^0] - P[B^1]\\ \text{ISC}(B^0 \Rightarrow A^1) &= P[A^1|B^0] - P[A^1]\\ \text{ISC}(A^1 \Rightarrow B^1) &= P[B^1|A^1] - P[B^1]\\ \text{ISC}(B^1 \Rightarrow A^1) &= P[A^1|B^1] - P[A^1]\end{aligned}$$

The first two lines give the interstimulus contingencies of a target appearing in one channel given that a target is not present in the other. Similarly, the third and fourth lines are the interstimulus contingencies of a target appearing in one channel given that a target is present in the other.

There are two commonly utilized task structures for the target detection DFP design that will exhibit ISC = 0 for all of the above ISCs of interest. In one structure, which we refer to as the *equal category rates design* we first consider the four trial types within the DFP design: redundant targets (A^iB^j where $i, j \in \{1,2\}$), single targets on the first source (A^iB^0 where $i \in \{1,2\}$), single targets on the second source (A^0B^j where $j \in \{1,2\}$), and target absent trails (A^0B^0). We set the base rate of presentation of each stimulus type to be equal, with $i, j \in \{1,2\}$:

$$P[A^iB^j] = P[A^iB^0] = P[A^0B^j] = P[A^0B^0] = .25.$$

This makes the unconditional probabilities of each target $P[A^i] = P[B^j] = .5$. The relevant conditional probabilities in this task, with $i, j \in \{1,2\}$, are

$$\begin{aligned}P[B^j|A^i] &= .5\\ P[A^i|B^j] &= .5.\end{aligned}$$

Thus, all the target ISC values are 0.

Note that in the equal category rate design, while the four types of trials are presented with equal likelihood, the rates for each individual stimulus in the design (i.e. the individual stimuli that incorporate the salience manipulations) are not equal. Rather, the rates for the individual stimuli depend on the number of stimuli that fall into the four trial categories; because there are different numbers of stimuli in each category, the individual rates vary by category. Assuming that the stimuli within a category are presented with equal likelihood, this design results in individual stimulus rates of:

$$\begin{aligned}P[A^2B^2] = P[A^2B^1] = P[A^1B^2] = P[A^1B^1] &= \frac{1}{16}\\ P[A^2B^0] = P[A^1B^0] = P[A^0B^2] = P[A^0B^1] &= \frac{1}{8}\\ P[A^0B^0] &= \frac{1}{4}.\end{aligned}$$

A second possible DFP design, which we refer to as the *equal stimulus rates design* starts by assuming that all nine individual stimuli in the DFP target detection design are presented equally, $P[A^iB^j] = \frac{1}{9}$ for $i, j = 0, 1, 2$, rather than basing the rate on the trial type categories. In this case, the unconditional probability of each target is $P[A^i] = P[B^j] = \frac{2}{3}$, and the relevant conditional probabilities, again with $i, j \in \{1,2\}$, are,

$$\begin{aligned}P[B^j|A^0] &= \frac{2}{3}\\ P[A^i|B^0] &= \frac{2}{3}\\ P[B^j|A^i] &= \frac{2}{3}\\ P[A^i|B^j] &= \frac{2}{3}\end{aligned}$$

Thus, the ISC values are all 0. Note that ISC = 0 holds if we break down all the probabilities by the stimulus level, high (2) or low (1), in which case all the unconditional and conditional probabilities are equal to $\frac{1}{3}$.

Potential ISCs in an experiment must be further balanced out with nontarget-driven decision biases that may be present in the task which can also be determined through contingency information (Mordkoff & Yantis, 1991). Where ISCs concern the statistical relationship between the identity (target/non-target) information from each source (and do not consider the assignment of each stimulus to a response rule), nontarget-driven decision biases are concerned with the influence that the presence of a non-target in the stimulus may have on biasing the decision process for a given task. These biases are captured in the computation of non-target response contingencies (NRCs). The NRC is computed as the difference between the conditional probability of a 'target present' response given a particular non-target item and the baseline proportion of 'target present' trials in the experiment. That is,

$$NRC(N) = P[+|N] - P[+]$$

where $P[+]$ is the unconditional probability of a 'target present' trial and $P[+|N]$ is the conditional probability of a 'target present' trial given the presence of a non-target stimulus item.

For the DFP target detection designs, we want to consider the NRCs for two stopping rules, OR and AND, using the two common designs outlined in the previous section. In the equal category rates design, in which $P[A^iB^j] = P[A^iB^0] = P[A^0B^j] = P[A^0B^0] = .25$, for the OR stopping rule:

$$\begin{aligned}P_{\text{OR}}[+] &= \frac{3}{4}\\ P_{\text{OR}}[+|A^0] = P_{\text{OR}}[+|B^0] &= \frac{1}{2}\\ \text{NRC}(A^0) = NRC(B^0) &= \frac{1}{2} - \frac{3}{4} = \frac{-1}{4}.\end{aligned}$$

And for the AND stopping rule:

$$P_{\text{AND}}[+] = \frac{1}{4}$$
$$P_{\text{AND}}[+|A^0] = P_{\text{AND}}[+|B^0] = 0$$
$$\text{NRC}(A^0) = \text{NRC}(B^0) = 0 - \frac{1}{4} = \frac{-1}{4}.$$

Thus for both stopping rules, we see by the negative NRC values that in the DFP detection design, the presence of a non-target trial is correlated with a 'target absent' response.

Now consider the equal stimulus rates design in which all nine stimuli are presented at equal rates. Starting again with the OR stopping rule:

$$P_{\text{OR}}[+] = \frac{8}{9}$$
$$P_{\text{OR}}[+|A^0] = P_{\text{OR}}[+|B^0] = \frac{2}{3}$$
$$\text{NRC}(A^0) = \text{NRC}(B^0) = \frac{2}{3} - \frac{8}{9} = \frac{-2}{9}.$$

And for the AND stopping rule:

$$P_{\text{AND}}[+] = \frac{4}{9}$$
$$P_{\text{AND}}[+|A^0] = P_{\text{AND}}[+|B^0] = 0$$
$$\text{NRC}(A^0) = \text{NRC}(B^0) = 0 - \frac{4}{9} = \frac{-4}{9}.$$

Again, for both the OR and AND decision rules, we see negative NRC values, implying that the presence of a non-target trial is correlated with a 'target absent' response.

Thus, we can see that for DFP target detection designs, whether your workload manipulation is target absense or the presence of a non-varying distractor, the DFP designs that exhibit $ISC = 0$ consistently produce an NRC that is is non-zero. It is critical, then, for NRC to be minimized in a DFP design, and as our results show, this can be dependent on the stopping rule chosen for the experiment. In Table 2, we highlight a good choice of stimulus rates for both the OR and AND stopping rules that will minimize NRC, according to the above computations.

In addition to stopping rule, the NRC value is also dependent on the total number of stimuli in an experiment. Consider the target/distractor discrimination task illustrated in Figure 13 in which the distractor arrows are also varying in their salience. This design contains the same four trial type categories (redundant targets, single targets on the first source (global arrows), single targets on the second source (local arrows), and target absence or, in this case, double distractors). However, a key difference between this design and the target detection with distractors in Figure 12 is the total number of stimuli; each category in this design contains four stimuli, for a total of 16 stimuli in the experiment instead of nine. It follows that the equal category rates design and equal stimulus rates design are essentially the same. Thus, for either an OR or AND decision rule, utilizing $P[A^iB^j] = \frac{1}{16}$ for $i, j = \pm1, \pm2$ as the likelihood of any individual stimulus will produce relevant ISC values of 0. Under this choice of stimulus rate, the $NRC = \frac{-1}{4}$ indicating a negative nontarget-decision bias in either type of task. Thus, for target/distractor discrimination with variable distractors, the stimulus rates that exhibit $ISC = 0$ and minimize NRC are the same for both the OR and AND tasks. We have listed this in Table 3, showing the same choice of stimulus rates for OR and AND, unlike Table 2 which has different recommendations for OR and AND tasks.

Importantly, across DFP designs that have been developed in the published literature, we know of no possible DFP stimulus rate structure in which the ISC values are all 0 and there is simultaneously an absence of nontarget-driven decision bias (i.e., NRC = 0). However, considering the above results, we can see that it may be possible to choose the stimulus rates so as to minimize the nontarget-driven decision bias. This choice may be dependent on both the choice of stopping rule and on the total number of stimuli in the experiment. For the OR stopping rule in the detection task with nine stimuli, the NRC value is lower for the equal stimulus rates design, where the presentation rates of all nine stimuli are equated. But for the AND stopping rule in the detection task with nine stimuli, the NRC value for the equal category rates design is lower, so equating the presentation of the stimulus types, rather than individual stimuli, will minimize the nontarget-driven decision bias. When the task switches to one with variable distractors, the OR and AND designs minimizing NRC utilize the same presentation rates. Bear in mind that if you choose to use or develop a DFP design for another type of task or decision rule (Same-Different, Go/No-Go, Categorization/Classification, etc.), you will want to utilize the above equations for ISC and NRC to optimize your stimulus presentation rates so as to minimize the possible biases in your design. Again, the reader is referred to Mordkoff and Yantis (1991) for additional details on stimulus contingencies.

The effect of ISC and NRC on performance and model interpretation can be determined by manipulating both types of contingencies and measuring the changes of the response times in the condition(s) of interest (like the redundant target condition). As of yet, theoretical work relating the influence of non-zero ISC or non-zero NRC values to the SFT measures is lacking, but is an important topic of future research.

DFP Data for sft

The **sft** package has additional tools that allow for analyzing multiple participants and conditions at once, as long as the data are in the proper format. The basic format is a data frame with six (or more) variables: Subject, Condition, Correct, RT, Channel1, Channel2, ..., Channel*n*. Each row in the data frame corresponds to a single trial. Subject and Condition are only used by the program to group data, so any coding scheme can be used. Correct is a logical variable indicating whether the participant correctly responded on that trial. RT is the time the participant took to respond on a given trial. Each Channel column indicates the level of the stimulus presented for a particular source. The possible values are ±2, ±1 or 0, where 2 indicates high salience, 1 indicates low salience and 0 indicates there was no information presented for that source. Positive values indicate that target information was presented on that source while negative values indicate distractor information. Because the focus of SFT is on analyzing multiple sources of information, at least two Channels are required, but any number are possible. The group SIC analysis in **sicGroup** is currently limited to two channels, but the capacity function, **capacityGroup**, is capable of analyzing data with more than two channels.

These variables will be sufficient to complete an SFT analysis of a DFP experiment with two targets/sources of information (i.e., two channels to be modeled). Additional variables may be included in your data frame, particularly any additional metadata like Gender, Age, Trial Number, Testing Session Number, etc. But these variables will be ignored by the **sft** functions.

Let us consider how the arrow stimuli from Figure 12 would be coded, as summarized in Table 2. For the global-local arrows, two channels are needed, with Channel1 coding the global arrow information and Channel2 coding the local arrow information. On either channel, the target absent condition (the dash pointing neither right nor left) is given the value 0; the high salience arrows are given a value of +2 and the low salience arrows are coded by +1. This same coding scheme can be utilized for any DFP design with target present/absent as the workload manipulation. Note that in Table 2, we have given the recommended coding scheme for the **sft** package as well as the recommended trial rates for minimizing NRCs in the designs while maintaining $ISC = 0$.

Now let us consider an alternative DFP design where the target absent condition is replaced by a distractor stimulus containing competing information. The recommended **sft** data coding and trial presentation rates are summarized in Table 3. For the arrows design in Figure 13, this means that when a target right-pointing arrow is not present, rather than a dash, the global or local arrow is a left-pointing arrow, containing distractor directional information. With two salience levels for both target and distractor stimuli, this design implements the ±1, ±2 coding scheme described above.

For an experimental design with more than two channels, n channels can be identified by creating a *Channel*α variable for each α in $\alpha = 1, 2, \ldots, n$. For example, in a 4-channel experiment, like the 4-letter words studied by Houpt and Townsend (2010a), the values on the channels should be coded in four variables: *Channel1*, *Channel2*, *Channel3*, *Channel4*. It is important that for however many channels are coded, the channel variables must be listed numerically and consecutively, following the naming convention outlined herein. Note also that as of this publication, only the capacity functions are capable of analyzing more than two channels; the SIC functions, **sic** and **sicGroup**, are capable of analyzing only the 2-channel DFP data. Extensions to additional numbers of factors (e.g., 2x2x2) or additional numbers of levels per factor (e.g., 3x3) are possible as the theory and measures can extend to *n* sources of information; we will add the appropriate extensions to the **sft** package as such theory is developed and published.

A set of sample data is included in the **sft** package, called **dots**. These data, from Eidels and Townsend (2009), include response time and accuracy from nine participants that completed two versions of the DFP task with the dots stimuli illustrated in Figure 11. Stimuli were either two dots, one above fixation and one below, a single dot above fixation, a single dot below fixation, or a blank screen. The salience manipulation was the contrast level of the dots, such that each dot was shown at either high or low contrast when present. In the OR decision task, participants were asked to respond "yes" if either of the dots was present; in the AND decision task, they were asked to respond "yes" only when both dots were present. More details of the task are available in Eidels and Townsend (2009) or Houpt and Townsend (2010b).

The **dots** data is a data frame containing the six required variables: *Subject*, *Condition*, *Correct*, *RT*, *Channel1*, *Channel2*. *Subject* is a character vector indicating the participant identifier code; here we have two letter codes for each of the nine participants. *Condition* is a character vector indicating which version of the task (OR or AND) the data are from. *Correct* is a logical vector indicating if the response on each trial was correct (TRUE or 1) or not (FALSE or 0). *RT* is the numeric vector of response times. *Channel1* and *Channel2* are numeric vectors indicating the stimulus level for the two dots, with *Channel1* giving the value for the upper dot and *Channel2* giving the value for the lower dot. There were no distractors in this task, so there are three possible values for *Channel1* and *Channel2*: 0 = target absent, 1 = target present at low contrast, 2 = target present at high contrast.

The rows in the **dots** data frame are the individual trials of the experiment. That is, all data from every trial by every participant is included. This is critical for estimating the full response time distributions for the SFT analyses.

Example 4: **dots**

```
# Load the dots data.
data(dots)
head(dots)
summary(dots)
```

			sft Data Coding		Trial Rates*	
Trial Type	Source 1	Source 2	Channel1	Channel2	OR	AND
Redundant Targets	High Target	High Target	2	2	$\frac{1}{9}$	$\frac{1}{16}$
	High Target	Low Target	2	1	$\frac{1}{9}$	$\frac{1}{16}$
	Low Target	High Target	1	2	$\frac{1}{9}$	$\frac{1}{16}$
	Low Target	Low Target	1	1	$\frac{1}{9}$	$\frac{1}{16}$
Single Target A	High Target	Absent	2	0	$\frac{1}{9}$	$\frac{1}{8}$
	Low Target	Absent	1	0	$\frac{1}{9}$	$\frac{1}{8}$
Single Target B	Absent	High Target	0	2	$\frac{1}{9}$	$\frac{1}{8}$
	Absent	Low Target	0	1	$\frac{1}{9}$	$\frac{1}{8}$
Target Absent	Absent	Absent	0	0	$\frac{1}{9}$	$\frac{1}{4}$

Table 2

Channel Codes for DFP Target Detection Designs

*Trial Rates are the proportion of the total trials recommended for minimizing the non-target response contingencies; see text for details. The OR task is based on the equal stimulus rate design and the AND task is based on the equal category rate design.

			sft Data Coding		Trial Rates*
Trial Type	Source 1	Source 2	Channel1	Channel2	OR & AND
Redundant Targets	High Target	High Target	2	2	$\frac{1}{16}$
	High Target	Low Target	2	1	$\frac{1}{16}$
	Low Target	High Target	1	2	$\frac{1}{16}$
	Low Target	Low Target	1	1	$\frac{1}{16}$
Single Target A	High Target	High Distractor	2	-2	$\frac{1}{16}$
	High Target	Low Distractor	2	-1	$\frac{1}{16}$
	Low Target	High Distractor	1	-2	$\frac{1}{16}$
	Low Target	Low Distractor	1	-1	$\frac{1}{16}$
Single Target B	High Distractor	High Target	-2	2	$\frac{1}{16}$
	High Distractor	Low Target	-2	1	$\frac{1}{16}$
	Low Distractor	High Target	-1	2	$\frac{1}{16}$
	Low Distractor	Low Target	-1	1	$\frac{1}{16}$
Target Absent / Redundant Distractors	High Distractor	High Distractor	-2	-2	$\frac{1}{16}$
	High Distractor	Low Distractor	-2	-1	$\frac{1}{16}$
	Low Distractor	High Distractor	-1	-2	$\frac{1}{16}$
	Low Distractor	Low Distractor	-1	-1	$\frac{1}{16}$

Table 3

Channel Codes for DFP Target-Distractor Discrimination Designs

*Trial Rates are the proportion of the total trials recommended for minimizing the non-target response contingencies while maintaining interstimulus contingencies equal to zero; see text for details.

```
# Calculate and analyze the SIC for each
# participant in each condition.
sicGroup(dots)

# Calculate Cor and Cand for each
# participant in each condition.
capacityGroup(dots)
```

We do not include the extensive output here. Instead, we encourage readers to install the **sft** package and familiarize themselves with the functionality using the **dots** data.

General Notes on Design

Since the emergence of the DFP in 1995, the design has been adapted from the original detection task (similar to Figure 11) to target discrimination (e.g., Johnson et al., 2010; Ingvalson & Wenger, 2005), same-different judgments (Perry, Blaha, & Townsend, 2008), categorization (Fifić et al., 2008;

Fifić & Townsend, 2010), memory search (Townsend & Fifić, 2004), and visual search (Fific, Townsend, & Eidels, 2008), and has been applied across a variety of stimuli such as alpha-numeric characters, faces, and novel visual objects. It is possible that with a good choice of workload and salience manipulations, any type of perceptual or cognitive decision making task on any class of stimuli can be adapted to a DFP design. Note that while we have illustrated the DFP with visual cognitive stimuli and tasks, DFP is not limited to the visual modality. For example, Altieri and Townsend (2011) utilized the DFP to analyze audio-visual speech perception using a design in which one source of information was the visual stimulus of a person mouthing a word and the other source of information was the auditory stimulus of the spoken word.

Whatever the task, stimulus class, and modality, there are some key things to keep in mind about the choices of both the workload and salience manipulations:

Workload Manipulation

- Workload refers to the number of target sources, which is sometimes the pure number of items in the display or the number of targets relative to a constant load of distractors.
- When using distractors, keep the number constant so that only the number of target sources is varying.
- When modeling both targets and distractors, have balanced conditions of fixed numbers of distractors relative to varying numbers of targets and fixed numbers of targets relative to varying numbers of distractors.
- Context, background or noise may be added to a stimulus, but should be held constant across all conditions in the design so as not to add another source of information competing with the target sources.
- The workload manipulation should be independent of or orthogonal to the salience manipulation so that the same physical manipulation is not being used to influence both the presence of a target and the speed of processing of that target.

Salience Manipulation

- The manipulations of stimuli to affect the relative processing speeds should exhibit effective selective influence, so that a single physical manipulation of the stimulus affects only the speed of one subprocess at a time.
- Salience is a relative manipulation in that the low salience manipulation should result in slower processing than is observed in the high salience condition. For example, a point of light near threshold is detected more slowly than a point of light well above threshold in its brightness or contrast. Thus, there is no absolute right or wrong amount/intensity of salience manipulation here, as long as the chosen factor levels result in an strong ordering of the response time distributions, $S_{high}(t) < S_{low}(t)$, without significantly lowering accuracy.
- Manipulations of target salience should be independent of the stopping rule or decision criteria for a given task.
- The salience manipulation should be independent of or orthogonal to the workload manipulation so that the same physical manipulation is not being used to influence both the presence of a target and the speed of processing of that target.

In our explication of the SFT measures, particularly SIC, we focused on the case of two targets or two processing channels, for both clarity and tractability and because most of the studies cited herein have utilized manipulations only two sources of information. Importantly, these measures are not limited to only two targets and are scalable to n sources for a given task. For an example of a three factor SFT study (2x2x2 factorial), see Perry et al. (2008). Work is currently under way to generalize the SIC to more sources of information. As these advances and others are published, they will be included in the R **sft** package.

Conclusion

Systems Factorial Technology has been applied in a wide variety of psychological studies, from basic psychophysical tasks (C.-T. Yang et al., 2011), visual cognition (Zehetleitner, Krummenacher, & Müller, 2009), aural cognition (Altieri & Townsend, 2011), memory (Townsend & Fifić, 2004), to more complex tasks in social psychology (H. Yang, Houpt, Khodadadi, & Townsend, 2011), developmental psychology (Von Der Heide, Wenger, Gilmore, & Elbich, 2011) and clinical psychology (Johnson et al., 2010; Neufeld, Townsend, & Jetté, 2007). Despite this diverse range of applicability, the use of SFT is fairly limited within psychology. One major factor contributing to its lack of use is the difficulty in translating from the complex mathematical derivations (Townsend & Nozawa, 1995; Townsend & Wenger, 2004; Houpt & Townsend, 2010b, 2012) to applied settings. By developing a package for the popular data analysis software R (R Development Core Team, 2011), we hope to alleviate this difficulty.

In addition to making the basic tools of SFT more accessible, the **sft** package can also serve as a repository for related code as new advances to the methodology arise. For example, forthcoming work on the use of functional principal components analysis with the capacity coefficient (Burns, Houpt, & Townsend, 2013) will soon be added to the **sft** package.

References

Aalen, O. O., Borgan, Ø., & Gjessing, H. K. (2008). *Survival and event history analysis: A process point of view*. New York: Springer.

Altieri, N., & Townsend, J. T. (2011). An assessment of behavioral dynamic information processing measures in audiovisual speech perception. *Frontiers in Psychology*, *2*, 1-15.

Blaha, L. M. (2010). *A dynamic Hebbian-style model of configural learning*. Unpublished doctoral dissertation, Indiana University, Bloomington, Indiana.

Brown, S. D., & Heathcote, A. (2008). The simplest complete model of choice response time: Linear ballistic accumulation. *Cognitive Psychology*, *57*, 153-178.

Burns, D. M., Houpt, J. W., & Townsend, J. T. (2013). Functional principal components analysis of workload capacity functions. *Behavior Research Methods*, 1-10. doi: 10.3758/s13428-013-0333-2

Burns, D. M., Pei, L., Houpt, J. W., & Townsend, J. T. (2009). *Facial perception as a configural process.* Poster presented at: Annual Meeting of the Cognitive Science Society.

Chechile, R. A. (2003). Mathematical tools for hazard function analysis. *Journal of Mathematical Psychology*, *47*, 478-494.

Chechile, R. A. (2011). Properties of reverse hazard functions. *Journal of Mathematical Psychology*, *55*, 203-222.

Donders, F. C. (1969). On the speed of mental processes. In W. G. Koster (Ed. & Trans.), *Acta psychologica* (p. 412-431). Amsterdam: North-Holland Publishing Company.

Donnelly, N., Cornes, K., & Menneer, T. (2012). An examination of the processing capacity of features in the Thatcher illusion. *Attention, Perception & Psychophysics*, *74*, 1475-1487.

Dzhafarov, E. N. (2003). Selective influence through conditional independence. *Psychometrika*, *68*, 7-26.

Dzhafarov, E. N., & Gluhovsky, I. (2006). Notes on selective influence, probabilistic causality, and probabilistic dimensionality. *Journal of Mathematical Psychology*, *50*, 390-401.

Dzhafarov, E. N., Schweickert, R., & Sung, K. (2004). Mental architectures with selectively influenced but stochastically interdependent components. *Journal of Mathematical Psychology*, *48*, 51-64.

Eidels, A., Donkin, C., Brown, S. D., & Heathcote, A. (2010). Converging measures of workload capacity. *Psychonomic Bulletin & Review*, *17*, 763-771.

Eidels, A., Houpt, J. W., Pei, L., Altieri, N., & Townsend, J. T. (2011). Nice guys finish fast, bad guys finish last: Facilitatory vs. inhibitory interaction in parallel systems. *Journal of Mathematical Psychology*, *55*, 176-190.

Eidels, A., & Townsend, J. T. (2009). *Testing response time predictions of a large class of parallel models within or and and redundant signals paradigm.* Presented at the 2009 Annual Meeting of the Society of Mathematical Psychology. Amsterdam, Netherlands.

Fifić, M., Little, D. R., & Nosofsky, R. M. (2010). Logical-rule models of classification response times: A synthesis of mental-architecture, random-walk, and decision-bound approaches. *Psychological Review*, *117*, 309-348.

Fifić, M., Nosofsky, R. M., & Townsend, J. T. (2008). Information-processing architectures in multidimensional classification: A validation test of the systems factorial technology. *Journal of Experimental Psychology: Human Perception and Performance*, *34*, 356-375.

Fifić, M., & Townsend, J. T. (2010). Information-processing alternatives to holistic perception: Identifying the mechanisms of secondary-level holism within a categorization paradigm. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *36*, 1290-1313.

Fific, M., Townsend, J. T., & Eidels, A. (2008). Studying visual search using Systems Factorial Methodology with target-distractor similarity as the factor. *Perception & Psychophysics*, *70*, 583-603.

Garner, W. R. (1974). *The processing of information and structure*. New York: Wiley.

Garner, W. R., & Felfoldy, G. L. (1970). Integrality of stimulus dimensions in various types of information processing. *Cognitive Psychology*, *1*, 225-241.

Houpt, J. W., & Townsend, J. T. (2010a). A new perspective on visual word processing efficiency. In S. Ohlsson & R. Catrambone (Eds.), *Proceedings of the 32nd annual conference of the cognitive science society* (p. 1148-1153). Austin, TX: Cognitive Science Society.

Houpt, J. W., & Townsend, J. T. (2010b). The statistical properties of the survivor interaction contrast. *Journal of Mathematical Psychology*, *54*, 446-453.

Houpt, J. W., & Townsend, J. T. (2011). An extension of SIC predictions to the Wiener coactive model. *Journal of Mathematical Psychology*, *55*, 267-270.

Houpt, J. W., & Townsend, J. T. (2012). Statistical measures for workload capacity analysis. *Journal of Mathematical Psychology*, *56*, 341-355.

Ingvalson, E. M., & Wenger, M. J. (2005). A strong test of the dual-mode hypothesis. *Perception & Psychophysics*, *67*, 14-35.

Johnson, S. A., Blaha, L. M., Houpt, J. W., & Townsend, J. T. (2010). Systems factorial technology provides new insights on global–local information processing in autism spectrum disorders. *Journal of Mathematical Psychology*, *54*, 53-72.

Link, S. W., & Heath, R. A. (1975). A sequential theory of psychological discrimination. *Psychometrika*, *40*, 77-105.

Miller, J. (1982). Divided attention: Evidence for coactivation with redundant signals. *Cognitive Psychology*, *14*, 247-279.

Mordkoff, J. T., & Yantis, S. (1991). An interactive race model of divided attention. *Journal of Experimental Psychology: Human Perception and Performance*, *17*, 520-538.

Neufeld, R. W., Townsend, J. T., & Jetté, J. (2007). Quantitative response time technology for measuring cognitive-processing capacity in clinical studies. In R. W. Neufeld (Ed.), *Advances in clinical cognitive science: Formal modeling and assessment of processes and symptoms* (p. 207-238). Washington, D. C.: American Psychological Association.

Perry, L., Blaha, L. M., & Townsend, J. T. (2008). Reassessing the architecture of same-different face judgments. *Journal of Vision*, *8*, 88. doi: 10.1167/8. 6.889

R Development Core Team. (2011). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria. Retrieved from `http://www.R-project.org` (ISBN 3-900051-07-0)

Ratcliff, R., & Smith, P. L. (2004). A comparison of sequential sampling models for 2-choice response. *Psychological Review*, *3*, 333-367.

Reinach, S. G. (1960). A nonparametric analysis for a multiway classification with one element per cell. *South African Journal of Agricultural Science*, *8*, 941-960.

Repperger, D. W., Havig, P. R., Reis, G. A., Farris, K. A., McIntire, J. P., Townsend, J. T., ... Houpt, J. W. (2009). Studies on hazard functions and human performance. *The Ohio Journal of Science*, *109*.

Sawilowsky, S. S. (1990). Nonparametric tests of interaction in experimental design. *Review of Educational Research*, *60*, 91-126.

Schwarz, W. (1989). A new model to explain the redundant-signals effect. *Perception & Psychophysics*, *46*, 498-500.

Schwarz, W. (1994). Diffusion, superposition, and the redundant-targets effect. *Journal of Mathematical Psychology*, *38*, 504-520.

Townsend, J. T. (1972). Some results concerning the identifiability of parallel and serial processes. *British Journal of Mathematical and Statistical Psychology*, *25*, 168-199.

Townsend, J. T. (1974). Issues and models concerning the processing of a finite number of inputs. In B. H. Kantowitz (Ed.), *Human information processing: Tutorials in performance and cognition* (p. 133-168). Hillsdale, NJ: Erlbaum Press.

Townsend, J. T., & Altieri, N. (2012). An accuracy-response time capacity assessment function that measures performance against standard parallel predictions. *Psychological Review*, *119*, 500-516.

Townsend, J. T., & Ashby, F. G. (1983). *The stochastic modeling of elementary psychological processes*. Cambridge: Cambridge University Press.

Townsend, J. T., & Eidels, A. (2011). Workload capacity spaces: A unified methodology for response time measures of efficiency as workload is varied. *Psychonomic Bulletin & Review*, *18*, 659-681.

Townsend, J. T., & Fifić, M. (2004). Parallel and serial processing and individual differences in high-speed scanning in human memory. *Perception & Psychophysics*, *66*, 953-962.

Townsend, J. T., & Honey, C. J. (2007). Consequences of base time for redundant signals experiments. *Journal of Mathematical Psychology*, *51*, 242-265.

Townsend, J. T., & Nozawa, G. (1995). Spatio-temporal properties of elementary perception: An investigation of parallel, serial and coactive theories. *Journal of Mathematical Psychology*, *39*, 321-360.

Townsend, J. T., & Thomas, R. D. (1994). Stochastic dependencies in parallel and serial models: Effects on systems factorial interactions. *Journal of Mathematical Psychology*, *38*, 1-24.

Townsend, J. T., & Wenger, M. J. (2004). A theory of interactive parallel processing: New capacity measures and predictions for a response time inequality series. *Psychological Review*, *111*, 1003-1035.

Von Der Heide, R. J., Wenger, M. J., Gilmore, R. O., & Elbich, D. (2011). Developmental changes in encoding and the capacity to process face information. *Journal of Vision*, *11*.

Yang, C.-T. (2011). Relative saliency in change signal affects perceptual comparison and decision processes in change detection. *Journal of Experimental Psychology: Human Perception and Performance*, *37*, 1708-1728.

Yang, C.-T., Chang, T.-Y., & Wu, C.-J. (2012). Relative change probability affects the decision process of detecting multiple feature changes. *Journal of Experimental Psychology: Human Perception and Performance*. (Advance online publication)

Yang, C.-T., Hsu, Y.-F., Huang, H.-Y., & Yeh, Y.-Y. (2011). Relative salience affects the process of detecting changes in orientation and luminance. *Acta Psychologica*, *138*, 377-389.

Yang, H., Houpt, J. W., Khodadadi, A., & Townsend, J. T. (2011). *Revealing the underlying mechanism of implicit race bias.* Poster presented at: Midwestern Cognitive Science Meeting. East Lansing, MI.

Zehetleitner, M., Krummenacher, J., & Müller, H. J. (2009). The detection of feature singletons defined in two dimensions is based on salience summation, rather than on serial exhaustive architectures. *Attention, Perception & Psychophysics*, *71*, 1739-1759.