Distribution Agreement

Signature:

______________________________ ________________

Saule Kulubekova Date

Computational Model of Selection by Consequences: Patterns of Preference Change on Concurrent Schedules

By
Saule Kulubekova
Doctor of Philosophy
Psychology

____
Jack J McDowell, Ph.D.
Advisor

____
Eugene Emory, Ph.D.
Committee Member

____
M. Jackson Marr, Ph.D.
Committee Member

____
Darryl Neill, Ph.D.
Committee Member

____
Phillip Wolff, Ph.D.
Committee Member

Accepted:

____
Lisa A. Tedesco, Ph.D., Dean of the James T. Laney School of Graduate Studies

____________________Date

Computational Model of Selection by Consequences:
Patterns of Preference Change on Concurrent Schedules

By

Saule Kulubekova
M.A., Emory University, 2006

Advisor: Jack J McDowell, Ph.D.

An abstract of
A dissertation submitted to the Faculty of the
James T. Laney School of Graduate Studies of Emory University
In partial fulfillment of the requirements for the degree of
Doctor of Philosophy in
Psychology
2012

Abstract

Computational Model of Selection by Consequences:

Patterns of Preference Change on Concurrent Schedules

By Saule Kulubekova

The computational model of selection by consequences is an ontogenetic dynamic account of adaptive behavior based on the Darwinian principle of selection by consequences. The model is a virtual organism based on a genetic algorithm, a class of computational algorithms that instantiate the principles of selection, fitness, reproduction and mutation. The computational model has been thoroughly tested in experiments with a variety of single alternative and concurrent schedules. A number of published reports demonstrated that the model generated patterns of behavior that were quantitatively equivalent to the findings from live organisms. The experiments and analyses proposed here assess the behavior of the computational model for evidence of preference change phenomena identified and described by Davison, Baum and their collaborators in concurrent schedule experiments with pigeons. Three types of preference change behavior observed in live organisms were tested in experiments with the virtual organism: molar and molecular effects of behavioral adjustment in rapidly changing environment and presence of preference pulses. The results of this study provide strong evidence in support of the selectionist account of adaptive behavior.

Computational Model of Selection by Consequences: Patterns of Preference Change on Concurrent Schedules

By

Saule Kulubekova
M.A., Emory University, 2006

Advisor: Jack J McDowell, Ph.D.

A dissertation submitted to the Faculty of the
James T. Laney School of Graduate Studies of Emory University
In partial fulfillment of the requirements for the degree of
Doctor of Philosophy in
Psychology
2012

Table of Contents

Tables

Figures

higher values for richer overall reinforcement rate. The plot on the left is 41
Figure 9 (p. 14) from Davison and Baum (2000), and is included for
comparison. The plot on the right is for the computational model.

Figure 6. Fits of Equation 2 for log response ratios prior to each successive 42
reinforcer versus arranged log reinforcer ratios. Values of sensitivity to
reinforcement increased progressively from close to zero prior to the first
reinforcer to 0.82 (RI 25 – filled circles) and 0.55 (RI 60 – unfilled circles)
prior to tenth reinforcer. Sensitivity to reinforcement was higher for the
higher overall reinforcement rate.

Figure 7. Log response ratios following each reinforcer delivery averaged 43
across all components. RI 25 and RI 60 are the two RI schedules. The plots
show all possible sequences up to the third reinforcer delivery. Solid lines
show left reinforcer deliveries and dotted lines show right reinforcer deliveries.
The plot on the left is Figure 15 (p. 18) from Davison and Baum (2000).

Figure 8. Log response ratios following selected sequences of left-right 44
reinforcer deliveries for conditions with overall reinforcement rates of
RI 25 and RI 60. Solid lines show *confirmations*, in which a reinforcer is
delivered for a response on the same alternative as for the previous reinforcer
delivery. Dotted lines show *disconfirmations*, which are reinforcer deliveries
for a response on an alternative different from the one previously reinforced.
On the left are two of the plots from Davison and Baum's (2000) paper (Figure
16 on p. 19).

Figure 9. Preference pulses for the seven reinforcer ratios. Following 45

reinforcement, preference was always for the alternative that delivered it.

On the left are the plots from Davison and Baum (2003) paper (Figure 8 on p. 107).

Computational Model of Selection by Consequences:

Patterns of Preference Change on Concurrent Schedules

The purpose of this study was to extend the evaluation of the computational model of selection by consequences, proposed by McDowell (2004), to a new domain of research on adaptive behavior. To explain the rationale for this project, it is necessary to present McDowell's computational model of selection by consequences, review extant research findings on his theory, and present the data from animal (pigeon) studies on preference change behavior patterns against which the behavior of the model will be evaluated in this study.

Computational Model of Selection by Consequences

The computational model of selection by consequences proposes an ontogenetic account of adaptive behavior. The model is a causal account of how behavior evolves in an organism in response to environmental change. In the past three decades mathematical description of behavior has become an important part of behavior analysis. A widely accepted and extensively validated mathematical account of operant behavior is matching theory (Baum, 1974, 1979; Davison & McCarthy, 1988; Herrnstein, 1970). Matching theory puts forth a set of equations describing the relationship between reinforcement and behavior. For behavior on concurrent schedules of reinforcement, the matching theory stipulates that the ratio of responses or time allocated to concurrent schedule alternatives follows the ratio of reinforcers obtained from the alternatives.

$$\frac{B_1}{B_2} = b\left(\frac{r_1}{r_2}\right)^a \quad , \tag{1}$$

where B_1, B_2, r_1, and r_2 are, respectively, the numbers of responses emitted and reinforcers obtained from alternatives 1 and 2. The parameters b and a represent deviations from perfect matching. The *b* parameter accounts for any bias for one alternative over the other. The parameter *a* represents sensitivity to reinforcement. In perfect matching situations, $a = 1$; undermatching is a tendency to respond in the direction of indifference toward consequences when $a < 1$; a tendency toward exclusive preference for a behavioral alternative is overmatching, $a > 1$. In standard concurrent schedule experiments, *a* values typically range between 0.7 and 1.0, with the most common values around being around 0.8 (Baum, 1974, 1979; Myers & Myers, 1977; Wearden & Burgess, 1982).

The relationship between reinforcement and behavior on concurrent schedules and the effects of bias and sensitivity parameters are shown in Figure 1 (McDowell, 1989). The three plots on the left show the effects of bias, as it appears in three coordinate systems. In the top panel, response proportions are plotted against reinforcement proportions; biased responding appears in the form of curves that bow away from the matching diagonal. In the middle panel, response rate ratios are plotted against reinforcement rate ratios; biased responding appears in the form of lines with varying slopes and constant intercepts equal to zero. In the bottom panel, logarithms of response rate ratios are plotted against logarithms of reinforcement rate ratios; biased responding appears in the form of lines with varying intercepts and constant slopes equal to unity. The heavy diagonal in all three panels represents perfect matching. The three plots on the right of the figure show the effects of undermatching. In the top panel, response proportions are plotted against reinforcement proportions; undermatching appears in the form of sigmoidal curves that deviate from the matching diagonal in the direction of indifference. In the middle panel, response rate ratios are plotted against reinforcement rate ratios; undermatching appears in the form of negatively accelerated curves that pass through the origin. In the bottom panel, logarithms of response rate ratios are plotted against common logarithms of reinforcement rate ratios; undermatching appears in the form of lines with varying slopes less than unity, and constant intercepts equal to zero.

Equation 1 provides a widely validated descriptive account of adaptive behavior on concurrent schedule of reinforcement. It describes *steady-state* behavior, or responding that is in equilibrium with conditions in the environment. The research on how behavior gets to the steady state has not yielded a generally accepted theory explaining why behavioral and environmental variables are related in the way described by the matching theory equations. Various analytical accounts of operant behavior have been proposed, such as maximization and melioration theories (Herrnstein, 1982; Rachlin, Battalio, Kagel, & Green, 1981; Vaughan, 1981); however, none has been universally accepted. Maximizing and melioration theories are broadly similar; an organism moves from decision point to a decision point, choosing the next step based on a prescribed algorithm. In maximization theory, the organism selects the next point with the largest value; in melioration theory, the organism selects the point for which all existing opportunities have equal values.

An alternative to the analytical approach is the computational approach. In a computational account, there is no prescribed outcome and no end state toward which behavior is directed. Instead, a set of low-level selection rules governs the relationship between behavior and its consequences, but there is no high-level condition that a behavior must satisfy.

McDowell’s (2004) computational account of adaptive behavior is based conceptually on the principle of selection by consequences, which originates in evolutionary biology (Skinner, 1981). The key implication is that the dynamics of selection by consequences drives the adaptation of organisms not only phylogenetically (evolution of species) but also ontogenetically (behavior adaptation during the life span of single organisms).

To evaluate the computational model of selection by consequences, McDowell (2004) developed a software application in which a virtual organism continuously emits behavior in an experimental environment. The model is based on a genetic algorithm, a class of computational algorithms used in artificial intelligence (AI) applications. Genetic algorithms are based on a set of principles originating in evolutionary biology, including selection, reproduction, fitness, and mutation. Potential outcomes or solutions exist as individuals in a population. The fitness function determines which individuals are “fit” to be selected for the next population. The population evolves as a result of the repeated application of the rules that instantiate the evolutionary principles.

The computational model also includes an experimental environment in which reinforcement schedules and other experimental parameters can be specified. The components and implementation of the computational model are summarized in Appendix A. The virtual organism behaves continuously in time, emitting one behavior from the population of potential behaviors each time tick. The emitted behavior is recorded and can be studied as if it were the behavior of a live organism. The steady-state performance takes place when the opposing forces of variation and selection reach a dynamic equilibrium in a given experimental setting (McDowell, 2004). Reinforcement has the effect of increasing the proportion of fitter behaviors in the population. Mutation and reproduction in the absence of reinforcement tend to have the opposite effect by introducing forces of variability and chance, which likely make the population of potential behaviors more diverse and less fit overall. At the point of dynamic equilibrium these forces generate a roughly constant rate of responding. The low-level rules of selection, reproduction, and mutation operate moment-to-moment and must be applied repeatedly to generate higher-level time-averaged equilibrium results.

The computational model is in the process if being tested which involves conducting studies that compare behavior of the model to the findings from experimental studies with various live organisms. A series of published studies that compared the behavior of the model to data from animal experiments are summarized in the next section.

Testing the Model (2004 – present)

The first series of experiments was conducted with single-alternative random interval (RI) schedules (McDowell, 2004). The model generated equilibrium response rates that were accurately described by the Herrnstein hyperbola, which is

$$R = \frac{kr}{r + r_e}, \tag{2}$$

where R represents response rate, r represents reinforcement rate, and k and r_e are parameters of the equation (Herrnstein, 1970; McDowell, 2004).

The model consistently showed a hyperbolic relation between response and reinforcement rates even when parameters of the model, such as the mutation rate and the mean of the fitness function, were varied over wide ranges. The hyperbolic fit was also consistently better than the fits provided by similar function forms (asymptotic exponential, asymptotic power function, and ramp function). McDowell (2004) concluded that repeated application of low-level rules of selection, reproduction, and mutation produced high-level quantitative order known to characterize the behavior of live organisms. In other words, the Herrnstein hyperbola was an emergent property of the selectionist dynamics.

The next major study investigated the behavior of the model on concurrent random interval (RI) schedules (McDowell, Caron, Kulubekova & Berg, 2008). The study found that the virtual organisms' steady-state behavior was well described by the power function matching (Equation 1). In addition, the parameters of the equation behaved in ways that were consistent with findings from experiments with live organisms. For example, in symmetrical concurrent schedules the bias parameter *b* varied around a value of approximately 1, which indicates an absence of bias toward either of the alternatives. In asymmetrical concurrent schedules, *b* tracked the magnitude of the asymmetry. The average exponent *a*, also known as sensitivity to reinforcement, was close to 0.8, which is a degree of undermatching that is typically found in experiments with live organisms (Baum, 1979; Myers & Myers, 1977; Wearden & Burgess, 1982). The data from the single-alternative (McDowell, 2004) and concurrent schedules (McDowell et al., 2008) studies indicate that the equations of matching theory may in fact be emergent properties of the evolutionary dynamics of selection by consequences.

The McDowell lab also pursued research that investigated whether the model produced behavior consistent with McDowell's (2005) version of the matching equations, which resolve the constant *k* violation of the matching theory (McDowell & Caron, 2007;

McDowell & Popa, 2010). The studies showed agreement with the power function versions of all the matching equations (McDowell, 1986).

Several published and unpublished studies were completed to investigate the effects of varying different parameters of the model. For example, Popa and McDowell (2009) reported that setting behavior class boundaries at points where adjacent behaviors are separated by large Hamming distances simulates effects of changeover delay (COD). The Hamming distance is the number of bits in the binary representation of two integers that do not match (explained further in Appendix B). Larger Hamming distances produced effects consistent with longer CODs.

The matching theory is a descriptive account of *steady-state* behavior. From this molar perspective, behavior is viewed as being shaped by the aggregation of experience. But the behavior of the model has also been evaluated from a more local perspective. Kulubekova and McDowell (2008) reported that the model's log survivor plots, a form of frequency distribution of inter-response times (IRTs), were similar in shape to plots from experiments with rats and pigeons. Log survivor plots did not show the "broken-stick" feature indicative of distinct bouts and pauses in responding that has been observed in some experiments with rats. The shape of the virtual organism's log survivor plots was more consistent with the data on reinforced responding in pigeons, in which the boundary between within-bout and between-bout responding is not as clear-cut.

In summary, the computational model produces a range of quantitative molar and molecular steady-state behavior patterns known to characterize behavior in live organisms. However, a number of key findings from animal experiments have not yet been studied in the virtual environment. To determine the limits of the model, tests must be extended to a broader domain of behavior.

Preference Change: Molar and Molecular Patterns

Over the past decade, Baum, Davison and their colleagues reported a number of related patterns in preference change for concurrent schedule alternatives. Their data come from concurrent schedule experiments with pigeons responding during a switching procedure similar to the procedure first introduced by Belke and Heyman (1994).

Davison and colleagues arranged several components per session with different reinforcer ratios associated with each. An overall VI schedule was arranged for the whole experiment. Within each component, two concurrent VI schedules were arranged dependently (Stubbs & Pliskoff, 1969), meaning that, whenever a reinforcer was arranged for one alternative, a reinforcer could not be arranged on the other alternative until the already arranged reinforcer was obtained. The sequence of the seven components in each session was determined randomly without replacement. The probabilities of reinforcers for each alternative were determined based on the reinforcer ratios chosen for the experiment.

Each component was followed by a 10-second blackout. The switching of concurrent schedules occurred rapidly, and without explicit discriminative stimuli. The procedure was designed to study changes in choice following rapid unpredictable changes in reinforcer ratio.

Davison and colleagues used the logarithmic version of Equation 1,

$$\log \frac{B_1}{B_2} = a \log \frac{r_1}{r_2} + \log b \,, \tag{2}$$

to investigate the degree of control that the change in reinforcer ratios has over preference in responding on one alternative versus the other. The sensitivity parameter, *a*, reflects the degree of change in response ratios resulting from a change in the reinforcer ratios (Lobb & Davison, 1975), thus sensitivity values for different environmental variables reflect the degree of control that those variables exert over preference. Preference change was assessed by plotting the logarithm of the ratio of responses on one key to responses on the other key as a function of successive reinforcers. Thus, when preference for one alternative is the same as for the other, $\log \frac{B_1}{B_2}$ is zero. This may happen when reinforcement rates are the same, such as in an RI 1 RI 1 schedule, and there is no inherent bias toward one of the alternatives. It is not uncommon for pigeons to have a bias for (respond more on) the right key.

The components in the rapid switching procedure lasted until a fixed number of reinforcers was obtained. The components were separated by short blackout periods. The next schedule was chosen randomly from a set of schedules selected for the experiment. The schedules were specified as the ratios of probabilities of reinforcement in the two components. For example, Davison and Baum (2000) used the seven concurrent schedule ratios (27:1, 9:1, 3:1, 1:1, 1:3, 1:9 or 1:27) in components that lasted until ten reinforcers were obtained. There was no discriminative stimulus to indicate which schedule was going to be presented next, although the rate of reinforcement possibly served as an implicit discriminative stimulus.

Davison and Baum’s experiments were designed to reveal molar and local processes that govern the transition between steady states. Their procedure and analyses allowed the study of a large number of transitions among different reinforcer ratios and the extraction of the essential features of these repeated transitions. The analyses also allowed them to magnify and study the effects of individual reinforcers and reinforcer sequences.

The experimental findings that were selected for evaluation in the virtual environment are summarized below in three groups: (1) molar features, (2) local effects of single reinforcers and different sequences of reinforcers, and (3) preference pulses.

Molar features of shifts in preference. Davison and Baum (2000) found that, after pigeons have been trained in the switching procedure, with each new component, their preference for a particular key shifted rapidly and leveled off after only six to eight reinforcers, which is significantly faster than what had been observed in other experimental procedures. Equation 2 can be fitted to response and reinforcer ratios to estimate the exponent a, also referred to as sensitivity to reinforcement, for each inter-reinforcer interval. Davison and Baum found that a reached relatively high values, about 0.6, quickly. Davison and Baum also found that sensitivity was higher when the overall rate of reinforcement for the session was higher, i.e. sensitivity values increased faster and reached higher values for the higher overall reinforcement rate. Davison and Baum concluded that acquisition of preference occurs much more rapidly than previously reported when the experimental procedure arranges rapid changes in environmental contingencies. It would be interesting to determine experimentally whether the virtual organism produces similar pattern of preference change and sensitivity values, and whether adaptation happens more quickly at higher overall rates of reinforcement.

Local Effects of Reinforcers. Davison and Baum (2000) examined the effects of different sequences of reinforcers on preference change by calculating response ratios in the inter-reinforcer intervals. At the beginning of a component, the average log response ratio is close to zero – no preference for either key. Each successive reinforcer produces a shift in preference toward the alternative from which it came. This can be represented as tree-like plots of average log response ratios for all possible reinforcer sequences. The greater "spread" of the trees reflects a wider range of preference changes. Preference shifts appear to be larger at higher overall reinforcement rates.

Another local effects analysis investigated effects of confirmations and disconfirmations. In Davison and Baum (2000), a *confirmation* was defined as reinforcer delivery on the same key that produced the previous reinforcer. A *disconfirmation* was defined as reinforcer delivery on the other key. Following each reinforcer delivery, preference shifts toward the just reinforced alternative, but effects of each successive reinforcer decrease when the source of reinforcement remains the same. In a plot of log response ratios as a function of delivered reinforcers, a series of confirmations appear as a curve ("branch") starting at close to zero prior to the first reinforcer, with the value of log response ratio initially increasing with each confirmation. After a number of reinforcers delivered on the same alternative, the effect of each successive confirmation decreases which is reflected in decreasing curvature of the two outermost branches (one for each alternative). In comparison, effects of disconfirmations continue to be substantial even after successive confirming reinforcers. This is reflected in a significant shift in the value of the log response ratio. A single disconfirming reinforcer still produces a large shift in preference even if it follows a long series of confirmations.

The influence of the overall reinforcement rate was also apparent. Higher overall reinforcement rates were associated with greater sensitivity values. This was also apparent in the plots of confirmations and disconfirmations. Higher log response ratios were reflected graphically in the greater spread of the plots at higher overall reinforcement rate.

Preference Pulses. Davison and Baum (2000) defined a preference pulse as a period of heightened preference for the concurrent schedule alternative that just produced a reinforcer. Preference pulse features are essentially local effects. A preference pulse can be plotted as preference change as a function of successive responses. The x-axis shows responses since the last reinforcer delivery. As in the earlier graphs, the y-axis shows changes in preference as log response ratio. A preference pulse appears as a large shift in preference toward just reinforced alternative immediately after reinforcer delivery. During the inter-reinforcer period preference for the last reinforced alternative tends to decline gradually in the direction of indifference.

In summary, the scope of this study samples a subset of preference change results from experiments with pigeons, starting with all experiments described in the first paper in this domain of research (Davison & Baum, 2000). These specific patterns were chosen because they comprise a diverse and representative set of molar, local, and intermediate molar-local patterns in preference change. These findings were also reproduced in several subsequent published studies, thus they are robust. The preference change patterns include Davison and Baum’s (2000) finding that, in rapidly changing environment, behavioral adjustment occurs very rapidly. The values of sensitivity to reinforcement (a in Equation 2) increase progressively and quickly from close to zero prior to the first reinforcer to the 0.5 – 0.7 range after only eight or nine reinforcers. Another pattern is that values of sensitivity to reinforcement increase faster at richer overall reinforcement rates. Part I will examine whether the virtual organism shows a similar rate of behavioral adjustment. Part II will test the effects of confirmations and disconfirmations. Part III will look for evidence of preference pulses, brief periods of extreme post-reinforcer preference, in the virtual organism's responding. The primary question for this study is whether these preference change patterns emerge as a property of selectionist principles in evolutionary dynamics.

Method

Subjects

The experiments were conducted using the virtual organism with a population of 100 potential behaviors at each moment, or tick, of time. The computational algorithm is described in detail in Appendix B.

Apparatus and Materials

Experiments were conducted on a Lenovo ThinkPad T400 with Intel Core™2 Duo CPU processor, running Windows Vista™ Ultimate. The software implementation of the model was developed by J. J McDowell in VB .NET, a programming language.

Procedure

Table 1 lists all the parameters of the model that can be manipulated experimentally. The parameter settings used in the current experiments were the same as in earlier concurrent schedule experiments (McDowell et al., 2008). This is a default set of settings used across several published studies. The same settings were used in order to show that no special fine tuning of the model's parameters was necessary to produce behavior patterns in this study. The following parameter specifications were selected: circular fitness landscape; midpoint fitness method; continuous selection method; linear selection function form; search matchmaking method; bitwise recombination method; bit flip by individual mutation method; 1024 possible behaviors (integers from 0 to 1023); 100 possible behaviors in the population at each time tick. The population was fully repopulated (100% replacement) at each time tick. The mean of the parental fitness function was 25. The mutation rate used was 10 %. This same set of parameter settings was used across all experiments for simplicity and consistency.

The study replicated the rapid switching procedure used by Davison and Baum (2000). Sessions were divided into seven components with a different concurrent schedule in each. The reinforcer ratios for the alternatives were 27:1, 9:1, 3:1, 1:1, 1:3, 1:9, 1:27, the same ratios used by Davison and Baum (2000). The order of the components was random without replacement. Each component was in effect until ten reinforcers were obtained.

An initial set of computational experiments was conducted in order to generate a representative sampling of the virtual organism's behavior during the switching procedure. These initial results were used to select the overall RI schedules.

Reinforcers were arranged dependently (Stubbs & Pliskoff, 1969), according to a single RI schedule. Once a reinforcer became available, it was allocated to one of the alternatives according to probabilities based on the seven reinforcer ratios listed earlier. No further reinforcers were allocated until the arranged reinforcer had been obtained.

Intervals for RI schedule were drawn randomly with replacement from an exponential distribution of intervals with a specified mean value. Hence the RI schedules were idealized Flesher-Hoffman (Fleshler & Hoffman, 1962) variable-interval (VI) schedules. A preliminary set of steady state experiments using RI schedules that ranged from RI 1 to RI 160 was conducted to select a rich and a lean overall reinforcement schedule, in order to assess the effect of the overall reinforcement rate on preference. Two representative RI schedules were selected, RI 25 and RI 60. In preliminary steady-state experiments, the largest values of the exponent *a* (Equation 3) were obtained for overall reinforcement rates delivered by schedules in the RI 25 – RI 30 range. RI 25 arranged a relatively rich overall schedule of reinforcement. RI 60 was selected as a relatively lean schedule of reinforcement.

Part I of the experiments focused on whether the virtual organism's behavior showed molar patterns described by Davison and Baum (2000). Each component ran until 10 reinforcers were obtained. Responses were categorized with respect to their occurrence in sequences of responses and reinforcers. These analyses required large pools of data to ensure that each category had a sufficient number of responses to generate reliable molar effects. To replicate their analyses, numbers of responses at each key were counted for the period from start of the component to the first reinforcer, then from the first reinforcer to the second and so on. Thus, the data were organized reinforcer by reinforcer, with responding measured prior to each reinforcer and after the previous reinforcer. These response counts were averaged across 50 presentations of each component. Equation 2 was then fitted to the averaged data to estimate sensitivity values (*a*). The estimates were used to assess whether *a* values fell in the expected range and whether (and after how many reinforcers) *a* reached the range of values reported by Davison and Baum.

Part II assessed more local effects of sequences of reinforcers for the virtual organism. In order to obtain enough occurrences of various sequences of reinforcers, each component ran until 40 reinforcers had been delivered. Davison and Baum's (2000) analyses were replicated as follows. Response ratios were calculated for each inter-reinforcer interval up to the fourth reinforcer. If L (left) and R (right) are concurrent schedule alternatives, there would be one response ratio from the beginning of a component until the first reinforcer (x = 0), two response ratios for the period between the first and second reinforcers (L and R), four response ratios after the second and before the third (LL, LR, RR, RL), and eight after the third and before the fourth (LLL, LLR, LRL, LRR, RLL, RLR, RRL, RRR). The features of virtual organism's preference change plots were qualitatively compared to Davison and Baum's findings with pigeons.

For Part III, mean log response ratios were plotted as a function of responses after reinforcer delivery in order to qualitatively test for the presence of preference pulses. Each component lasted until 10 reinforcers had been delivered. Log response ratios were plotted for each response following reinforcement on each of the two alternatives for each of the seven components. Two hundred sessions were run to try to ensure sufficient number of responses to calculate the log response ratio at each ordinal position following a reinforcer. Response ratios were calculated by summing the number of responses at each ordinal position after a reinforcer delivery across all presentations of a component.

Results

Part I

The first analysis investigated how preference, as reflected in log response ratios, changed as a function of successive reinforcers. Following Davison and Baum's (2000) analysis, the responses were counted from reinforcer to reinforcer, that is, this organization was not cumulative. Figure 2 shows the relationship between log response ratio and successive reinforcers in each of the seven components. To create these plots, all responses from the start of the component to the first reinforcer were pooled across all 50 presentations of the component, and then all responses were pooled from the first reinforcer to the second, from the second to the third, and so on.

The average response ratio was close to zero before the first reinforcer (0 on the *x* axis), reflecting no preference for either alternative at the start of sessions. For L:R ratio of reinforcers, log response ratio, *log(L/R)*, values were positive when responding favored the L alternative; log response ratio values were negative when responding favored the R alternative; neither alternative was favored when L = R, for 1:1 ratio of reinforcers. Absolute values of log response ratios increased rapidly during the initial two to four reinforcer deliveries. The log response ratio appeared to asymptote as more reinforcers were delivered.

Figure 2 also shows the impact of reinforcer ratio. Response ratios changed with increasing number of reinforcers in a pattern fitting the arranged reinforcer ratio. Log response ratios increased faster and reached higher values in components with greater discrepancy between alternatives. For example, during the first few reinforcer deliveries, preference changed more rapidly in components with 1:27 and 27:1 reinforcer ratios, in comparison to responding in components with smaller ratios, such as 1:3 or 9:1 reinforcer ratios. This was consistent with Davison and Baum's (2000) findings. The curvature of the plots also reflected greater preference shifts following reinforcers from the richer alternatives.

Figure 3 includes an analogous plot from Davison and Baum (2000), for comparison. The plot on the right was constructed based on the data produced by the computational model (same as Figure 2). The plot on the left is Figure 7 from Davison and Baum (2000, p. 11). The two plots are remarkably similar, both qualitatively and quantitatively. The seven curves in their figure show preference change as a function of delivered reinforcers for experiments with different numbers of reinforcers per component, ranging from 4 to 12, and for two overall reinforcement rates, 2.22 reinforcers per minute and 6 reinforcers per minute. Davison and Baum reported that the number of reinforcers per component had no apparent impact on changes in preference. The computational model experiments therefore used the same setting of 10 reinforcers per component for all experiments in Part I. The curves in Davison and Baum's figure appear smoother than the plots for the computational model. However, these plots presented group data aggregated across six pigeons, with 35 sessions of data for each. The computational model's plots show data for 50 sessions with one virtual organism. The left side of Figure 4 shows data for a single pigeon, Bird 91 in Davison and Baum (their Figure 1 on p. 5). On the right is the plot for the computational model. Note the similarity in log response ratio values as well as the similarly more jagged nature of the plots.

Figure 2 also shows changes in preference for two overall reinforcement rates, RI 25 (filled circles) and RI 60 (unfilled circles). Numbers of responses at each key were counted for each inter-reinforcer interval. Thus, the data were organized reinforcer by reinforcer, with responding measured prior to each reinforcer and after the previous reinforcer, and averaged across 50 presentations of each component. Preference changed more rapidly and reached higher values for the richer overall reinforcement rate. This finding was also consistent with Davison and Baum's (2000) results. Figure 5 illustrates the influence of overall reinforcement rate more distinctly. Equation 2 was fitted to log response ratios prior to each successive reinforcer versus arranged log reinforcer ratios. The plot shows that sensitivity to reinforcement was higher for the higher overall reinforcement rate, RI 25. Values of sensitivity to reinforcement increased quickly from close to zero prior to the first reinforcer to relatively high values. After the fourth reinforcer, sensitivity reached 0.78 for RI 25 and 0.60 for RI 60. The plot on the left is Figure 9 (p. 14) from Davison and Baum (2000), included for comparison. The two curves for *12 R/C* and *12 R/C Hi* show the same pattern as the curves for RI 25 and RI 60. *Hi* indicates the condition with higher reinforcement rate and *12 R/C* means 12 reinforcers per component.

Figure 6 shows sensitivity to reinforcement as a function of successive reinforcers in all conditions of the experiment. For this plot, as in Davison and Baum (2000), the responses were pooled for inter-reinforcer intervals across all components. Each plot of the figure shows log response ratio as a function of log reinforcer ratio for each of the ten inter-reinforcer intervals, pooled across all components and sessions. Between the start of a component and the first reinforcer delivery, sensitivity (exponent a) was close to zero. Sensitivity to reinforcement increased rapidly with each reinforcer delivery. The two patterns of preference change described earlier are also apparent in this analysis: a faster increase in sensitivity values for the higher overall reinforcement rate, and quick progression (after only 4-5 reinforcer deliveries) of sensitivity to reinforcement to values comparable to those obtained in steady-state experiments. Both patterns are consistent with Davison and Baum's findings.

Part II

Part II focused on more local effects of reinforcers by examining response ratios in the inter-reinforcer intervals after various sequences of reinforcers. For this analysis, as in Davison and Baum (2000), the sequences of reinforcers were aggregated across various component reinforcer ratios. The frequency of each sequence therefore varied depending on the component reinforcer ratios. For example, for the 1:27 ratio, a sequence of reinforcers on the left would be very unlikely. Responses were also pooled across all 50 presentations of each component. To determine the impact of overall reinforcement rate, these analyses were done separately for the two RI schedules.

Response ratios were calculated for all possible sequences up to the third reinforcer. Thus, the response ratio was calculated at the beginning of a component, up to the first reinforcer delivery. For the second inter-reinforcer interval, two response ratios were calculated: one following a reinforcer on the left (L) and the other following a reinforcer on the right (R). For the third inter-reinforcer interval, four response ratios were calculated, one for each of the four sequences of two reinforcers: LL, LR, RR, and RL. Finally, for the fourth inter-reinforcer interval, there were eight possible sequences of three reinforcers for the two alternatives.

Figure 7 shows the plot of all possible response ratios up to the third reinforcer. On the left is a portion of an analogous plot from Davison and Baum (2000, two bottom graphs from Figure 15, p. 18). The right side of the figure shows the plots for the computational model, for RI 25 and RI 60. The plots have the shape of a tree, starting at 0 on the *x* axis and branching wider and outward with each reinforcer delivery. Before the first reinforcer, the average response ratio was close to zero, reflecting no preference at the beginning of a session. Preference, as reflected in the value of the log response ratio, shifted with each successive reinforcer toward the alternative that delivered it. Solid lines show shifts in preference after reinforcer deliveries on the left; dotted lines show shifts in preference after reinforcer deliveries on the right.

When each successive reinforcer was delivered on the same alternative, that is, a confirmation, preference continued to shift toward that alternative. The effects of such successive same-alternative reinforcers decreased, which is reflected in the falling curvature of the outermost branches of the plot. In contrast, the effect of disconfirmations was dramatic, resulting in significant shifts in preference toward the alternative that delivered the disconfirming reinforcer.

The effect of the overall reinforcement rate can be seen in the spread of the trees. For the higher overall reinforcement rate, RI 25, the spread of the tree plot was greater, indicating that preference shifted more with successive reinforcers in that condition.

Figure 8 highlights the effects of confirmations and disconfirmations showing them up to the eighth reinforcer. The plots show sequences of right only and left only reinforcers obtained in succession and the effects of a single disconfirmation at each sequential position. The left side of Figure 8 includes the plots from Davison and Baum’s (2000) paper (their Figure 16 on p. 19). In a sequence of confirmations, the effects of successive confirming reinforcers decreased. Disconfirmations dramatically shifted preference toward the reinforced alternative. Unlike confirmations, the effects of disconfirmations did not appear to decrease even when a disconfirmation followed a sequence of successive confirmations. The plots for pigeons and the computational model are very similar. Both have tree-like shape, with outer branches reflecting preference shifts after successive confirmations; both also reflect large preference shifts after disconfirming reinforcers. The plots are also quantitatively similar. Log response ratios for both pigeons and the model fell in the same range of values.

The two plots for RI 25 and RI 60 in Figure 8 also show the effect of overall reinforcement rate. The two plots in the lower half of the figure are wider reflecting greater shifts in preference in experiments with richer overall reinforcement rate (12 R/C Hi for pigeons and RI 25 for the model).

Part III

Following the progression from a molar to a more molecular perspective, Part III focused on preference pulses, examining response-level effects of reinforcers. The right side of Figure 9 shows these patterns for the computational model. Similar plots from Davison and Baum (2003; their Figure 8 on p. 107) are included on the left side of Figure 9, for comparison. For the computational model, log response ratio was plotted up to the fiftieth response following a reinforcer, prior to another reinforcer delivery. To make the plots, data were aggregated across two hundred sessions in order to collect enough across-session responses at each ordinal position following a reinforcer. At one on the *x* axis is the log response ratio at the first response after a reinforcer. The largest number of responses was aggregated across sessions at earlier ordinal positions, i.e. at lower *x*. Moving to the right on the *x* axis, fewer and fewer responses could be included in the calculation. This is because few inter-reinforcement intervals were that long. As a result, estimates of log response ratio at higher ordinal positions are less precise than immediately after a reinforcer. Very few to no responses occurred on the leaner alternative at higher ordinal response positions. This was also dependent on the reinforcer ratio arranged. For example, in 27:1 components, there were no responses on the lean alternative after 16 ordinal response positions; thus, log response ratio could not be calculated from that position on. Graphically, this resulted in truncated plots of log response ratio.

Comparison of the left and right sides of Figure 9 reveals both similarities and differences in responding of the computational model and pigeons. Consistent with live organisms, preference shifted toward the alternative that delivered a reinforcer, resulting in a spike in log response ratio toward the reinforcing alternative immediately after the reinforcer. Also consistent with Davison and Baum’s (2003) findings, the plots show that preference shifts were greater when reinforcers were delivered on the richer alternative. For example, for the 1:27 and 27:1 ratios, immediately after a reinforcer on the leaner alternative, responding favored that alternative about 4-fold over the non-reinforced richer alternative. When a reinforcer was delivered on the richer alternative, responding favored that alternative 12 to 13-fold over the non-reinforced leaner alternative in these components. The plot for components with 1:1 ratio of reinforcers is nearly symmetrical, suggesting that responding favored reinforced alternatives equally in these components.

A number of differences are quite apparent, however. Preference eventually and gradually moved toward indifference in all components in experiments with pigeons. For the computational model, this appears to happen only for the richer alternatives in 1:27, 27:1, 1:9, and 9:1 components. For the computational model, preference appeared to remain on the just reinforced alternative in components in which the arranged reinforcement rates were more similar, such as in components with 1:1, 1:3, and 3:1 ratios. Another difference is that for leaner alternatives, there were fewer occurrences when preference was maintained on the just reinforced alternative. This is probably due to another reinforcer occurring, likely on the other (richer) alternative, which began another inter-reinforcer interval.

Discussion

Investigations of choice behavior in highly variable environments have shown that preference can change very quickly. This study examined whether the computational model of selection by consequences produced preference change patterns observed in live organisms using Davison and Baum's (2000) procedure which arranged rapidly changing concurrent schedules. The first series of experiments examined preference changes within components depending on the number of reinforcers delivered. Davison and Baum found that sensitivity values reached about 0.6 after only six to eight reinforcers. In the computational model experiments the results were very similar. Sensitivity reached the same range of values rapidly and appeared to stabilize after only four to five reinforcers (Figure 5).

However, comparison with Davison and Baum's (2000) results, which are included on the left side of Figure 5, indicates that there were differences as well. The model appeared to demonstrate higher sensitivity to reinforcement, as reflected in higher values of exponent *a* after fewer reinforcers. Sensitivity to reinforcement can be manipulated by experimental parameters, such as characteristics of the experimental apparatus and the magnitude of the reinforcer. In the computational model experiments, this can also be a consequence of experimental settings, such as the mean of the parental fitness function and mutation rate. Some of these setting may be analogous to parameters of real-life experiments. For example, McDowell (2004) showed that the mean of the parental fitness function can represent the magnitude of reinforcement. In the experiments that produced data for Figure 5, the mean of the parental fitness function was set to 25. A higher mean produces a less strict selection rule, and has an effect similar to a lower magnitude of reinforcement. This, in turn, would likely produce lower values of exponent *a* and a slower rate of adaptation.

Another parameter that may influence sensitivity to reinforcement is mutation rate. A higher mutation rate introduces a greater amount of noise or randomness in the populations of potential behaviors, which leads to lower sensitivity to reinforcement (McDowell, 2004; McDowell et al., 2008). A more thorough investigation of how parameters of the model influence adaptation in the rapid switching procedure is an interesting topic for future study.

Figure 5 suggested another difference between the responding of pigeons and the computational model. The model's sensitivity to reinforcement reached high values and began to asymptote after about four reinforcers, which is when the exponent, *a*, began approaching 0.8. Moving toward an asymptote was not apparent in Davison and Baum's (2000) plots, reproduced on the left side of Figure 5. In their experiments, the values of sensitivity to reinforcement began to approach 0.8 only around tenth reinforcer. In all likelihood, an asymptotic trend would have become apparent after more reinforcer deliveries for Davison and Baum's pigeons.

The impact of overall reinforcement rate was found to be consistent with Davison and Baum's (2000) findings. Sensitivity to reinforcement reached higher values for the richer overall reinforcement rate. This pattern can be seen in Figure 5. The plot shows that values of sensitivity to reinforcement increased quicker when overall reinforcement schedule was RI 25, compared to RI 60.

The extent of preference change was also affected by the extent of the difference in reinforcement between alternatives. Consistent with Davison and Baum’s (2000) results, preference changed more rapidly and sensitivity reached higher values when the difference in reinforcement rate between the alternatives was greater. It is interesting that this feature was an emergent property of selection by consequences, that is, the feature was not explicitly implemented. In experiments with live organisms, especially humans, this would more likely be given a cognitive (as opposed to a behaviorist) attribution. For example, the discrepancy in ratio of reinforcers implicitly signals change in component schedules, which is more detectable when the discrepancy is greater.

To summarize, the experiments in Part I showed directional and quantitative effects of reinforcers on behavior. The directional effects of each reinforcer could be predicted based on the alternative at which it was delivered. Quantitatively, sensitivity to reinforcement (*a* in Equation 2) increased progressively and quickly from close to zero prior to the first reinforcer to the range of values comparable to those obtained in steady-state experiments. Sensitivity to reinforcement increased faster at richer overall reinforcement rates. These effects were consistent with reported findings from experiments with pigeons (Davison & Baum, 2000).

Part II shifted focus toward more local effects, specifically the effects of individual reinforcers as a function of their position in a sequence of reinforcers. The computational model produced tree-like plots (Figures 7 and 8) similar to the plots in Davison and Baum’s (2000) paper. Successive confirmations at an alternative continued to shift preference toward that alternative. The effect of confirming reinforcers appeared to decline with each reinforcer. Disconfirmations, on the other hand, produced dramatic shifts in preference. Effects of disconfirming reinforcers did not appear to decline even when the disconfirming reinforcer followed a series of confirmations.

Also consistent with Davison and Baum’s (2000) findings were the effects of overall reinforcement rate. Higher overall reinforcement rate produced wider tree plots, indicating stronger preference shifts on the richer schedules.

There were also a few differences in the features of the plots for the computational model when compared to plots for Davison & Baum's (2000) pigeons. For the pigeons, if disconfirmation occurred after about the fourth reinforcer, it moved preference to approximate indifference (plots on the left of Figure 8). For the computational model, disconfirmations continued to shift preference past indifference and strongly toward the alternative that delivered the reinforcer, as reflected in the reversal of the sign of the logarithm of the response ratio (plots on the right side of Figure 8). This was more apparent for the leaner overall reinforcement schedule where disconfirmations shifted preference to the level consistent with a series of confirming reinforcers at that alternative, as reflected in disconfirmation preference shifts that nearly reached the outermost branches of the tree plots. The plots of the model also appear less “orderly” and symmetric compared to the results from pigeons. These features may be due to the model’s parameter settings in these experiments, such as mutation rate and percent of possible behaviors replaced in the population at each time step. These features may also be a result of experimental parameters, such as the number of reinforcers delivered in each component and the set of reinforcer ratios used in the experiments.

Nonetheless, there is potential for at least several future studies to explore how parameters of the model and environment influence characteristics of adaptive behavior.

In summary, in the experiments with the computational model, reinforcer effects varied depending on reinforcer position in a series of reinforcers and whether a reinforcer was a confirmation or a disconfirmation. If several successive reinforcers were obtained from the same alternative, each reinforcer had diminishing effects on behavior in all experiments. In contrast, disconfirming reinforcers in such sequences of same-alternative reinforcers had a comparatively large impact on preference. These preference change patterns were consistent with the findings from experiments with pigeons (Davison & Baum, 2000).

Examination of preference pulse patterns for the computational model in Part III revealed both similarities and differences when compared to the behavior of live organisms. In the computational model experiments, immediately after reinforcer delivery, log response ratio spiked toward the reinforcing alternative. Preference shifts after reinforcers were stronger for alternatives with higher reinforcement rates, and weaker for alternatives with lower reinforcement rates. Both patterns were consistent with Davison and Baum's (2003) findings.

Preference for the just reinforced alternative was expected to decline with time between reinforcers, as was shown in Davison and Baum's (2003) plots. Their plots, included on the left side of Figure 9, suggest that this happened for all components and reinforced alternatives. For the computational model, this decline in preference was apparent only for rich schedule alternatives, i.e. after reinforcers on the left side for 27:1 and 9:1, and on the right side for 1:27 and 1:9 components. Preference appeared to change little (or even increased toward the just reinforced alternative) with successive responses. Some of the plots for the computational model also were truncated. This happened when no responses occurred on the non-reinforced alternative and/or on the leaner reinforced alternative, even across two hundred sessions, for ordinal response positions after a certain number of responses. One reason for this may be a very long COD that made switching alternatives too prohibitive. A future study can examine how Hamming distance at the boundary between target classes, which may be the computational model's equivalent of a COD (Popa & McDowell, 2010), influences characteristics of preference pulses. A possible solution may be aggregation of responses at each ordinal position across many more sessions, until log response ratios can be calculated at each response position.

In conclusion, this study provided an initial look at whether preference change patterns observed in experiments with pigeons emerge as a property of selectionist principles in evolutionary dynamics. The study fully reproduced experiments in Davison and Baum's (2000) study and examined one of the preference pulse patterns, similar to the analysis in Davison and Baum's (2003) paper. A number of intriguing questions remain for future research. Some of the topics for future investigation are how parameters of the model, such as the mean of the parental fitness function, mutation rate, target class boundaries (which would affect the COD analog), influence the impact of individual reinforcers.

Davison and Baum (2000) noted that, given the regularities evident at a molar level of analysis, it has been assumed, often implicitly, that behavior is controlled by relatively large aggregations of reinforcers. They proposed using the switching procedure to study behavior change in rapidly changing environments in order to investigate the effects of individual reinforcers, subtitling their paper “Every reinforcer counts.” Their findings showed that the effects of individual reinforcers differed in a number of ways depending on the context in which they were obtained. This context included factors that had effects on both the molar and the local level, such as the alternative from which the previous reinforcer was obtained and the number and sequence of successive reinforcers obtained from that alternative. The studies by Davison, Baum and their colleagues showed that each reinforcer had reliable directional and quantitative effects on behavior.

The sample of experiments in this study demonstrated analogous forces shaping behavior generated by the computational model. The results also showed that variables controlling choice operated at both molar and local levels. Successive continuing reinforcers had cumulative effects on choice. Various sequences of continuations and discontinuations had effects both at and within inter-reinforcer intervals. Individual reinforcers directly shaped subsequent behavior. Importantly, the model did not implement any of these behavior patterns explicitly. Instead, these features of adaptive behavior were emergent properties of selection by consequences. The low-level selection rules produced both the local effects of individual reinforcers, the molar effects of aggregations of reinforcers that were reported in previous studies (McDowell, 2004; McDowell, Caron, Kulubekova & Berg, 2008), as well as the patterns at various intermediate molar/local levels of analyses. These results provide strong evidence in support of the selectionist account of adaptive behavior.

References

Baum, W. M. (1974). On two types of deviation from the matching law: Bias and undermatching. *Journal of the Experimental Analysis of Behavior, 22*, 231-242. doi: 10.1901/jeab.1974.22-231

Baum, W. M. (1979). Matching, undermatching, and overmatching in studies of choice. *Journal of the Experimental Analysis of Behavior, 32*, 269-281. doi: 10.1901/jeab.1979.32-269

Baum, W. M., & Davison, M. (2004). Choice in a variable environment: Visit patterns in the dynamics of choice. *Journal of the Experimental Analysis of Behavior*, *81*, 85–127. doi: 10.1901/jeab.2004.81-85

Belke, T. W. & Heyman, G. M. (1994). Increasing and signaling background reinforcement: Effect on the foreground response–reinforcer relation. *Journal of the Experimental Analysis of Behavior*, *61*, 65–81. doi: 10.1901/jeab.1994.61-65

Davison, M., & McCarthy, D. (1988). *The matching law: A research review*. Hillsdale, NJ: Erlbaum.

Davison, M. & Baum, W. M. (2000). Choice in a variable environment: Every reinforcer counts. *Journal of the Experimental Analysis of Behavior, 74*, 1–24. doi: 10.1901/jeab.2000.74-1

Davison, M., & Baum, W. M. (2002). Choice in a variable environment: Effects of blackout duration and extinction between components. *Journal of the Experimental Analysis of Behavior, 77*, 65–89. doi: 10.1901/jeab.2002.77-65

Davison, M., & Baum, W. M. (2003). Every reinforcer counts: Reinforcer magnitude and local preference. *Journal of the Experimental Analysis of Behavior, 80*, 95–129. doi: 10.1901/jeab.2003.80-95

Fleshler, M., & Hoffman, H. S. (1962). A progression for generating variable-interval schedules. *Journal of the Experimental Analysis of Behavior*, *5*, 529-530. doi: 10.1901/jeab.1962.5-529

Herrnstein, R. J. (1970). On the law of effect. *Journal of the Experimental Analysis of Behavior, 13,* 243–266. doi: 10.1901/jeab.1970.13-243

Herrnstein, R. J. (1982). Melioration as behavioral dynamism. In M. L. Commons, R. J. Herrnstein, & H. Rachlin. (Eds.), *Quantitative analyses of behavior, Vol. 2: Matching and maximizing accounts*. Cambridge, MA: Ballinger.

Krägeloh, C. U., Davison, M., & Elliffe, D. M. (2005). Local preference in concurrent schedules: The effects of reinforcer sequences. *Journal of the Experimental Analysis of Behavior*, *84*, 37–64. doi: 10.1901/jeab.2005.114-04

Kulubekova, S. & McDowell, J. J (2008). A computational model of selection by consequences: Log survivor plots. *Behavioural Processes*, *78*, 291-296. doi: 10.1016/j.beproc.2007.12.005

Landon, J., & Davison, M. (2001). Reinforcer-ratio variation and its effects on rate of adaptation. *Journal of the Experimental Analysis of Behavior*, *75*, 207–234. doi: 10.1901/jeab.2001.75-207

Lobb, B. & Davison, M. C. (1975). Performance in concurrent interval schedules: A systematic replication. *Journal of the Experimental Analysis of Behavior, 24,* 191–197. doi: 10.1901/jeab.1975.24-191

McDowell, J. J (1986). On the falsifiability of matching theory. *Journal of the Experimental Analysis ofBehavior, 45*, 63-74. doi: 10.1901/jeab.1986.45-63

McDowell, J. J. (1989). Two modern developments in matching theory. *The Behavior Analyst*, *12*, 153–166.

McDowell, J. J. (2004). A computational model of selection by consequences. *Journal of the Experimental Analysis of Behavior*, *81*, 297-317. doi: 10.1901/jeab.2004.81-297

McDowell, J. J, Soto, P. L., Dallery, J., & Kulubekova, S. (2006). A computational theory of adaptive behavior based on an evolutionary reinforcement mechanism. In M. Keijzer (Ed.), Proceedings of the 2006 Conference on Genetic and Evolutionary Computation (GECCO-2006) (pp. 175-182). New York: ACM Press.

McDowell, J. J & Caron, M. L. (2007). Undermatching is an emergent property of selection by consequences. *Behavioural Processes*, *75*, 97-106. doi: 10.1016/j.beproc.2007.02.017

McDowell, J. J, Caron, M. L., Kulubekova, S., & Berg, J. P. (2008). A computational theory of selection by consequences applied to concurrent schedules. *Journal of the Experimental Analysis of Behavior*, *90*, 387-403. doi: 10.1901/jeab.2008.90-387

McDowell, J. J & Popa, A. (2010). Toward a Mechanics of Adaptive Behavior: Evolutionary Dynamics, Matching Theory Statics. *Journal of the Experimental Analysis of Behavior, 94*, 241–260. doi: 10.1901/jeab.2010.94-241

Myers, D., & Myers, L. (1977). Undermatching: A reappraisal of performance on concurrent variable-interval schedules of reinforcement. *Journal of the Experimental Analysis of Behavior, 27*, 203-214. doi: 10.1901/jeab.1977.27-203

Popa, A., & McDowell, J.J (2010). The effect of Hamming distances in a computational model of selection by consequences. *Behavioural Processes, 84,* 428-434. doi: 10.1016/j.beproc.2010.02.002

Rachlin, H., Battalio, R., Kagel, J., & Green, L. (1981). Maximization theory in behavioral psychology. *Behavioral and Brain Sciences, 4,* 371–417. doi: 10.1017/S0140525X00009407

Russell, S., & Norvig, P. (2003). *Artificial intelligence: A modern approach*. Saddle River, NJ: Prentice-Hall.

Stubbs, D. A., & Pliskoff, S. S. (1969). Concurrent responding with fixed relative rate of reinforcement. *Journal of the Experimental Analysis of Behavior, 12,* 887–895. doi: 10.1901/jeab.1969.12-887

Shull, R. L., & Grimes, J. A. (2003). Bouts of responding from variable interval reinforcement of lever pressing by rats. *Journal of the Experimental Analysis of Behavior, 80,* 159-171. doi: 10.1901/jeab.2003.80-159

Vaughan, W., Jr. (1981). Melioration, matching, and maximization. *Journal of the Experimental Analysis of Behavior, 36,* 141–149. doi: 10.1901/jeab.1981.36-141

Wearden, J. H., & Burgess, I. S. (1982). Matching since Baum (1979). *Journal of the Experimental Analysis of Behavior, 38,* 339-348. doi: 10.1901/jeab.1982.38-339

Table 1

Parameters of McDowell's Computational Model of Selection by Consequences[1]

Feature	Variants	Standard
Number of generations to be run	Infinite	20,000
Number of generations discarded for shaping purposes	Limited by generations run	500
Size of behavioral set	Infinite	1,024
Size of behavioral repertoire	Limited by set size	100
Repertoire topography	Circular, flat	Circular
Percentage of repertoire to replace	0-100%	100%
Number of behavioral classes	Limited by repertoire size	4
Number of target classes	Limited by number of classes	0 (no reinforcement) 1 (single-alternative) 2 (concurrent schedule)
Selection schedule	Constant probability, random ratio, random interval	Random interval
Number of schedules	Infinite	11
Probability, mean ratios, or mean intervals	0-100, or infinite	1, 2, 3, 5, 10, 18, 25, 68, 112, 200 time ticks
Base probabilities	Must total to 1.00	0.46, 0.04, 0.04, 0.46
Fitness method	Midpoint, individual, class specific individual	Midpoint

---

[1] The table was created by Marcia L. Caron, who gave the author permission to include it in this paper.

Parental fitness function (PFF)	Linear, uniform, reciprocal, exponential	Linear
Mean of the PFF	Limited by set size	10, 20, 40, 100, 200
Reproduction method	Bitwise, slicewise, average, Gaussian (must specify *SD*)	Bitwise
Mutation method	Gaussian (must specify SD), bitflip by individual, bitflip by bit, random individual	Bitflip by individual
Mutation boundary	Wrap, discard	Wrap
Mutation rate	0-100%	1, 3, 5, 10, 20, 50
Changeover delay	Infinite	0
Selection method	Continuous, crop (must specify percentage), tournament (must specify number of competitors)	Continuous
Selection properties	Echo, persistence without mutation, persistence with mutation	None
Echo or persistence duration	Infinite	0 ticks
Echo or persistence function form	Linear, exponential, parabolic rectangular, sigmoid (must specify a parameter)	None

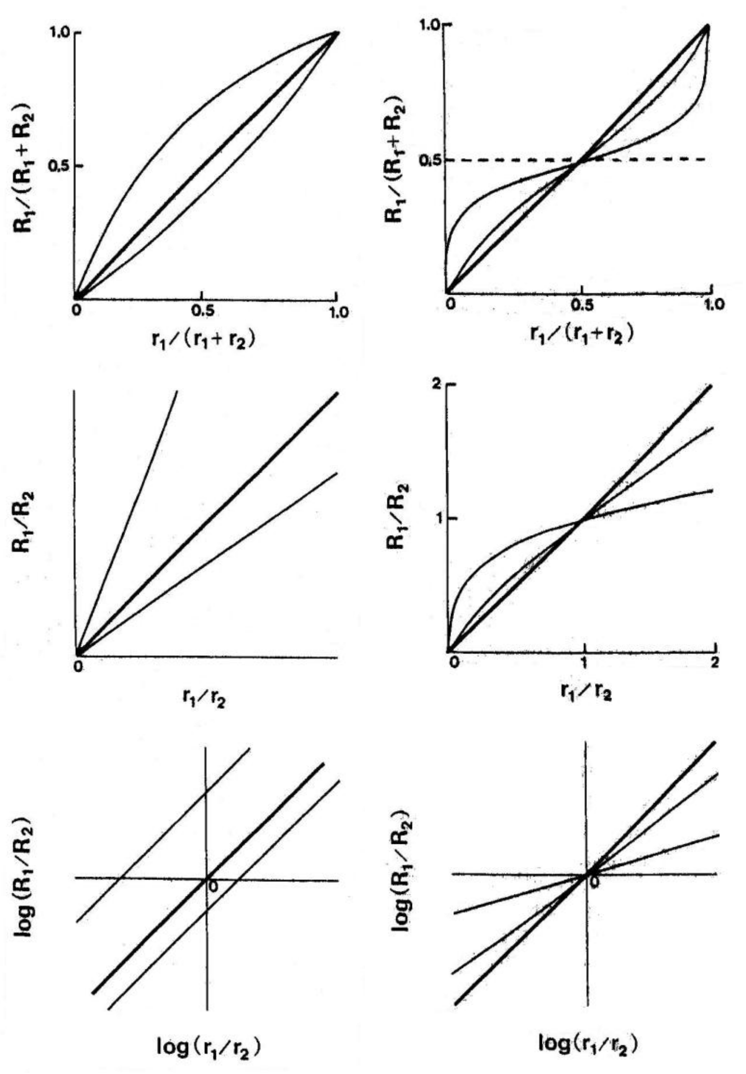

Figure 1. The heavy diagonal in all three panels represents perfect matching. The three graphs on the left show effects of the *bias* parameter in three coordinate systems. The three graphs on the right show effects of the *sensitivity parameter* (undermatching or overmatching) in three coordinate systems. Indifferent responding is represented by the dashed horizontal line. [Figure reprinted with permission of author (McDowell, 1989, pp. 157, 162).]

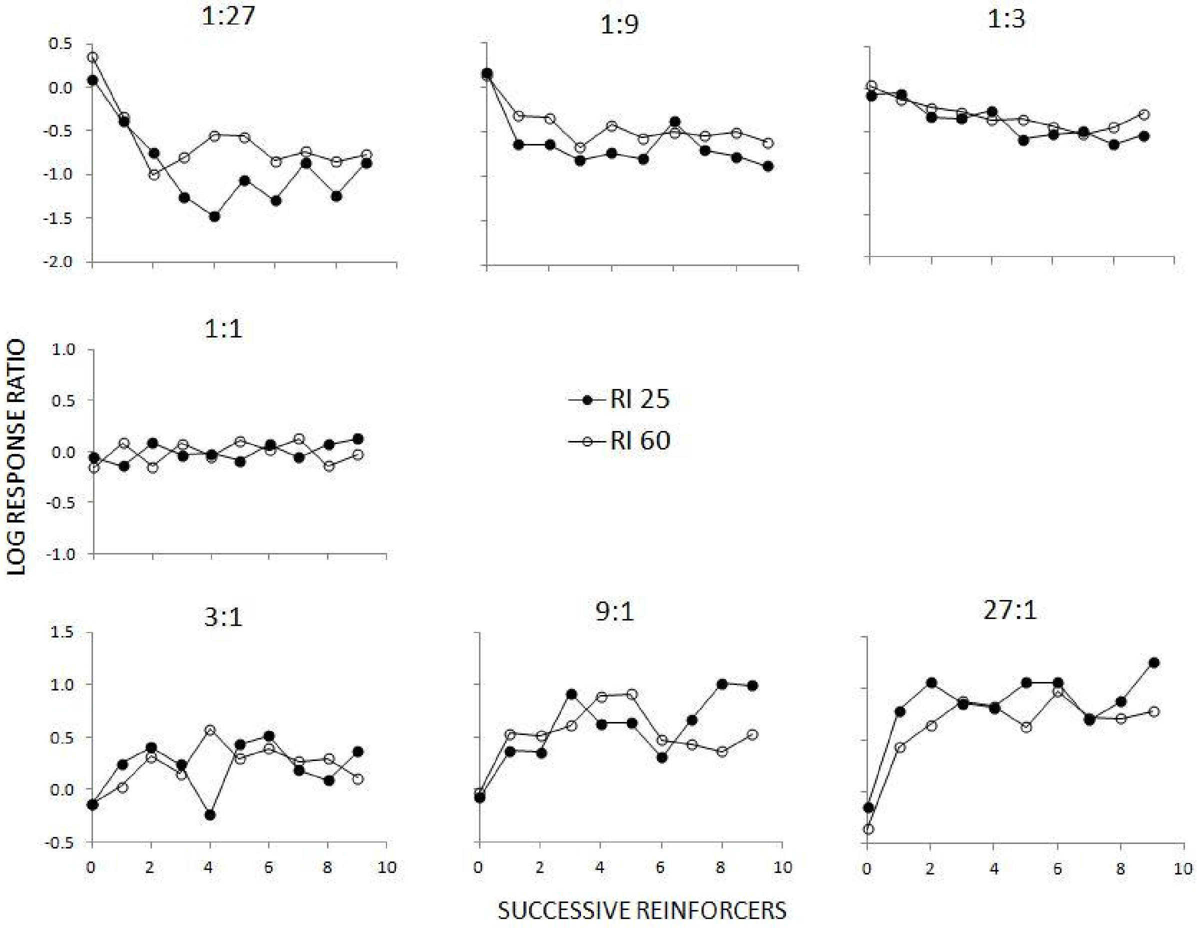

Figure 2. Log response ratio as a function of successive reinforcers. In components with unequal reinforcer ratios, the change in response ratios was greater when there was a greater contrast in reinforcement between the alternatives.

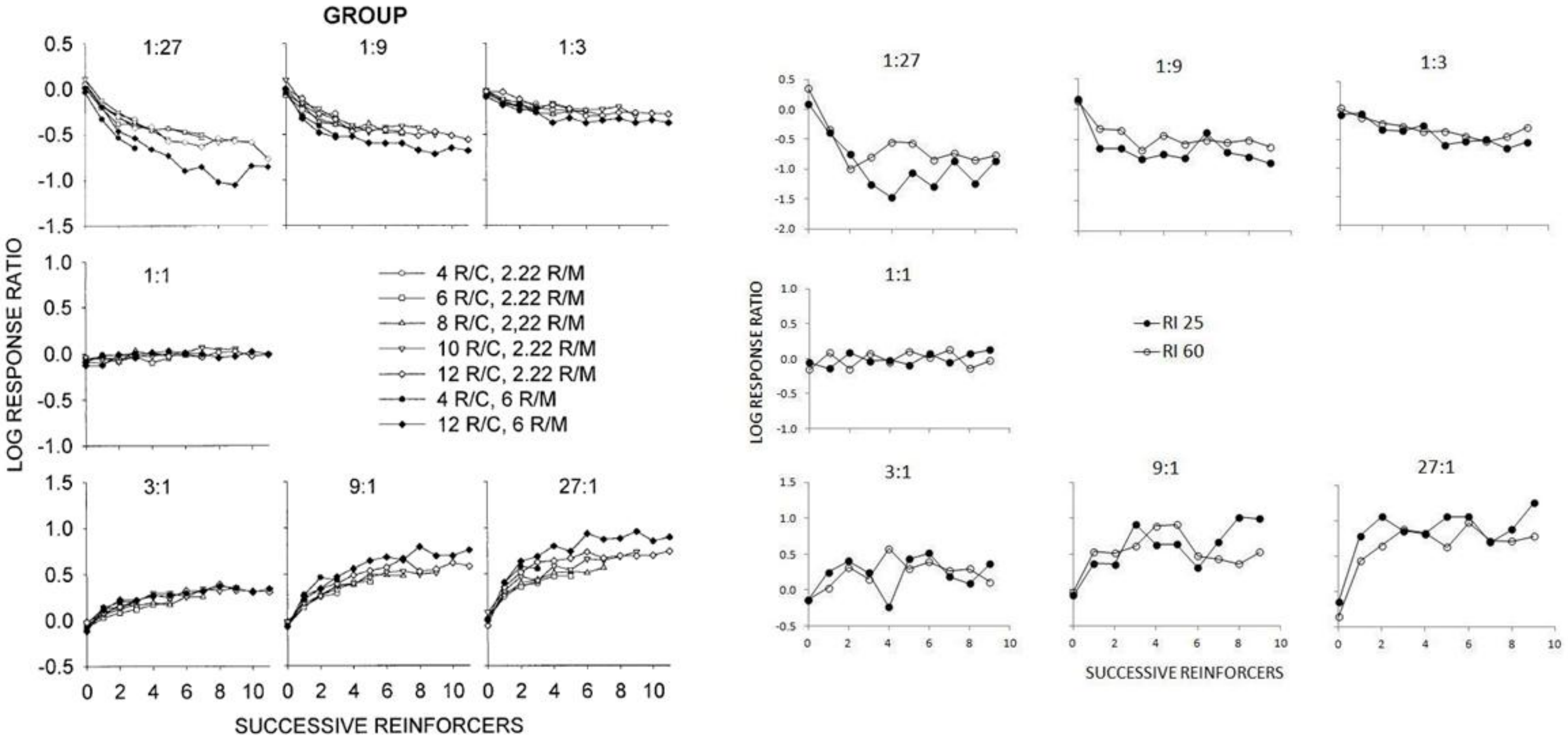

Figure 3. Log response ratio as a function of successive reinforcers. In components with unequal reinforcer ratios, the change in response ratios was greater when there was a greater contrast in reinforcement between the alternatives. The plots on the left are from Davison and Baum (2000; Figure 7, p. 11).

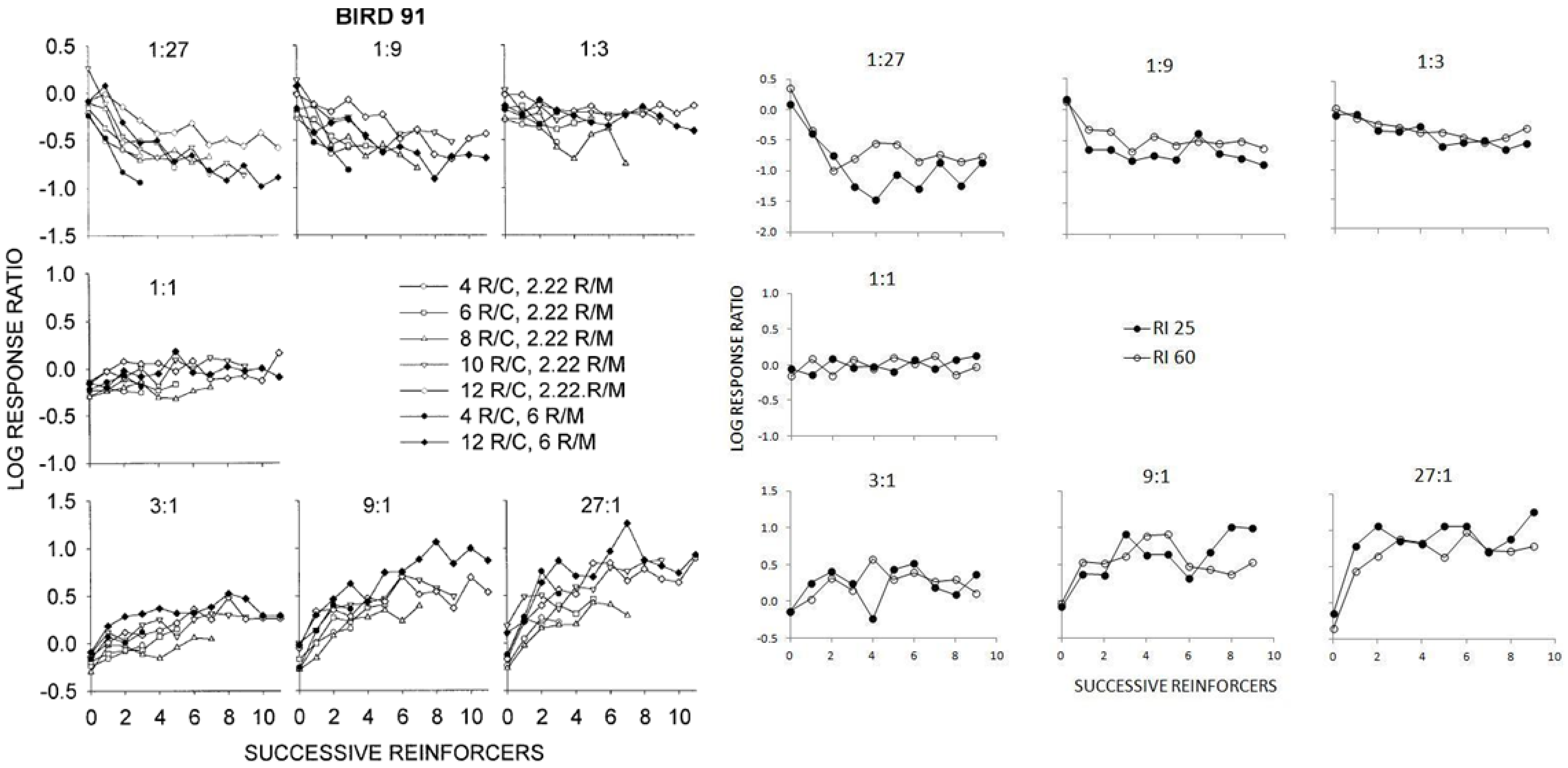

Figure 4. Log response ratio as a function of successive reinforcers. In components with unequal reinforcer ratios, the change in response ratios was greater when there was a greater contrast in reinforcement between the alternatives. On the left side are plots for a single pigeon, Bird 91 from Davison and Baum (2000, Figure 1, p. 5).

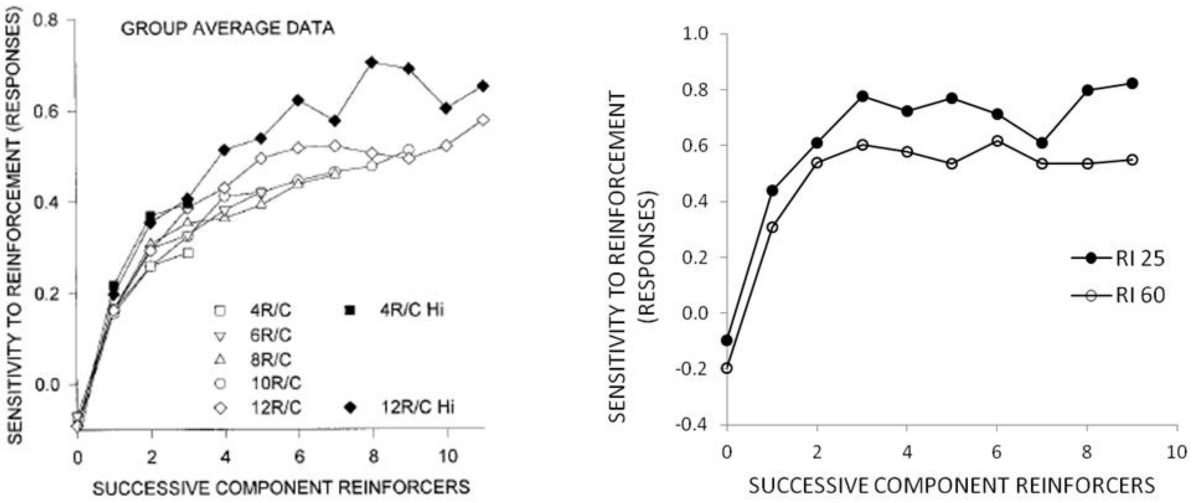

Figure 5. Sensitivity to reinforcement (*a* in Equation 2) as a function of successive reinforcers. Sensitivity values increased faster and reached higher values for richer overall reinforcement rate. The plot on the left is Figure 9 (p. 14) from Davison and Baum (2000), and is included for comparison. The plot on the right is for the computational model.

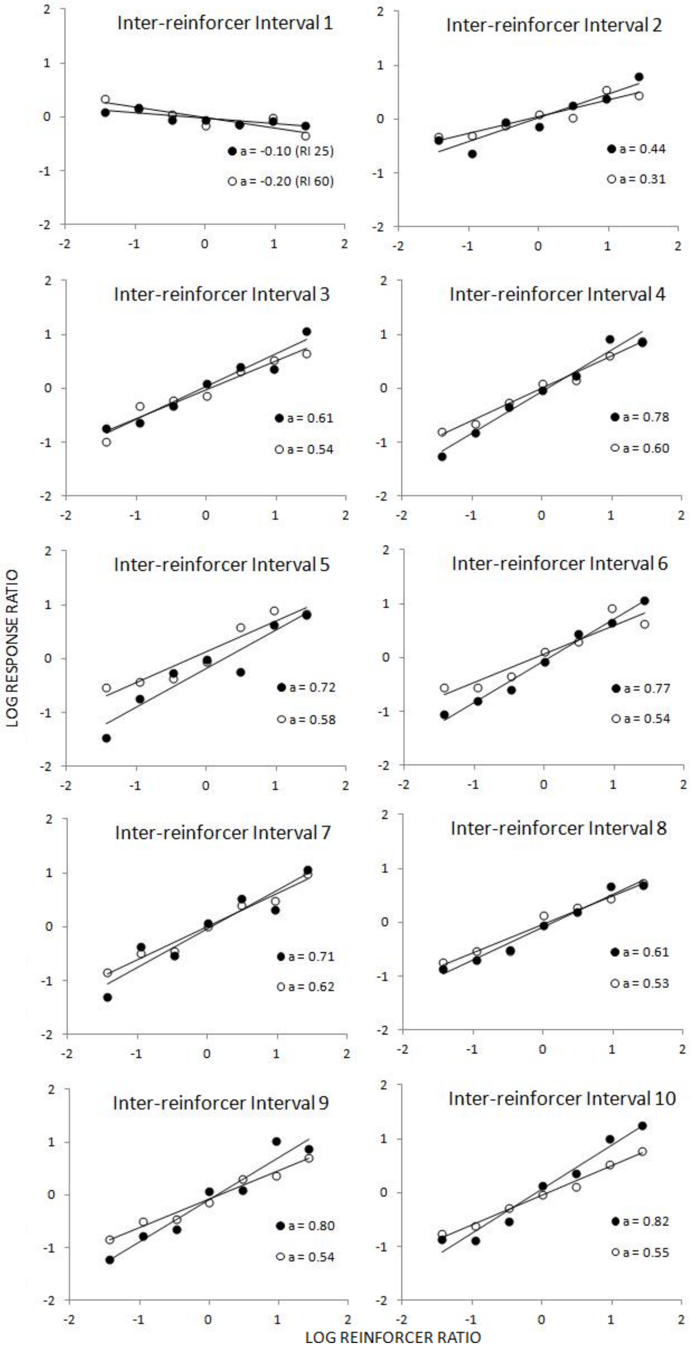

Figure 6. Fits of Equation 2 for log response ratios prior to each successive reinforcer versus arranged log reinforcer ratios. Values of sensitivity to reinforcement increased progressively from close to zero prior to the first reinforcer to 0.82 (RI 25 – filled circles) and 0.55 (RI 60 – unfilled circles) prior to tenth reinforcer. Sensitivity to reinforcement was higher for the higher overall reinforcement rate.

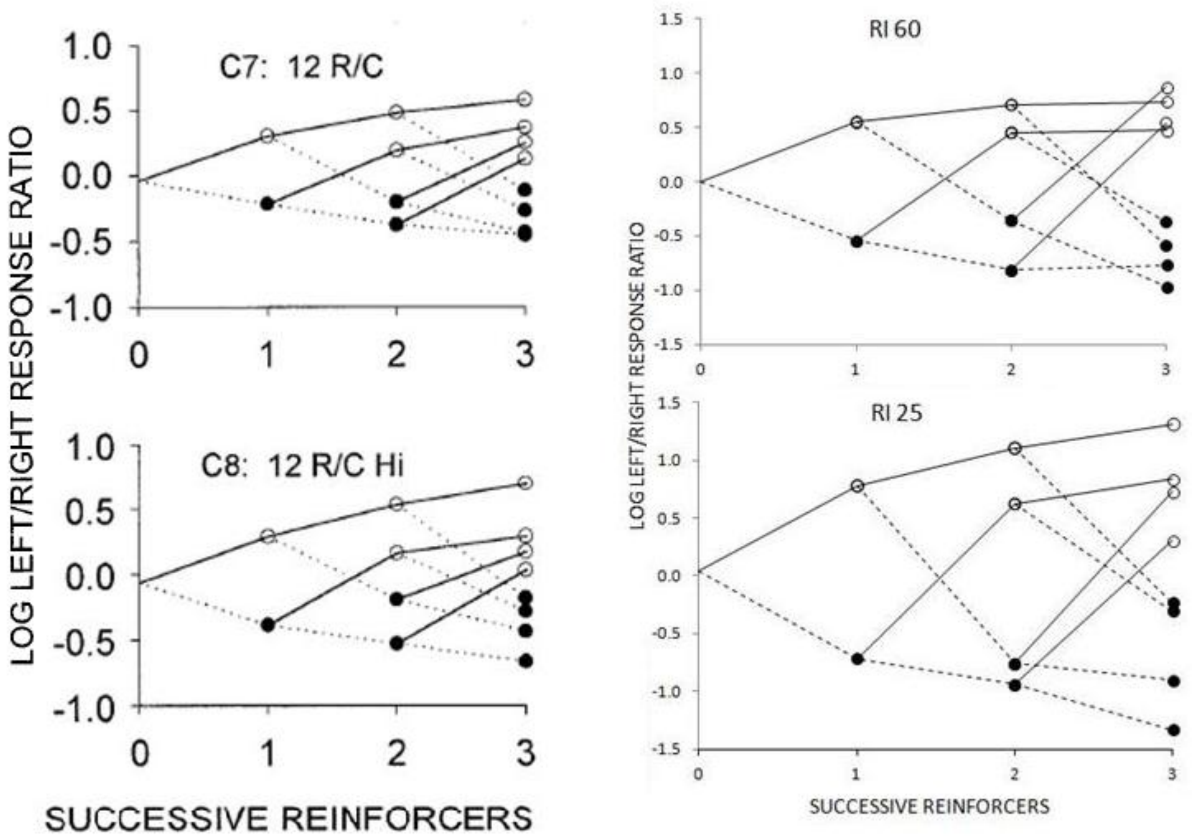

Figure 7. Log response ratios following each reinforcer delivery averaged across all components. RI 25 and RI 60 are the two RI schedules. The plots show all possible sequences up to the third reinforcer delivery. Solid lines show left reinforcer deliveries and dotted lines show right reinforcer deliveries. The plot on the left is Figure 15 (p. 18) from Davison and Baum (2000).

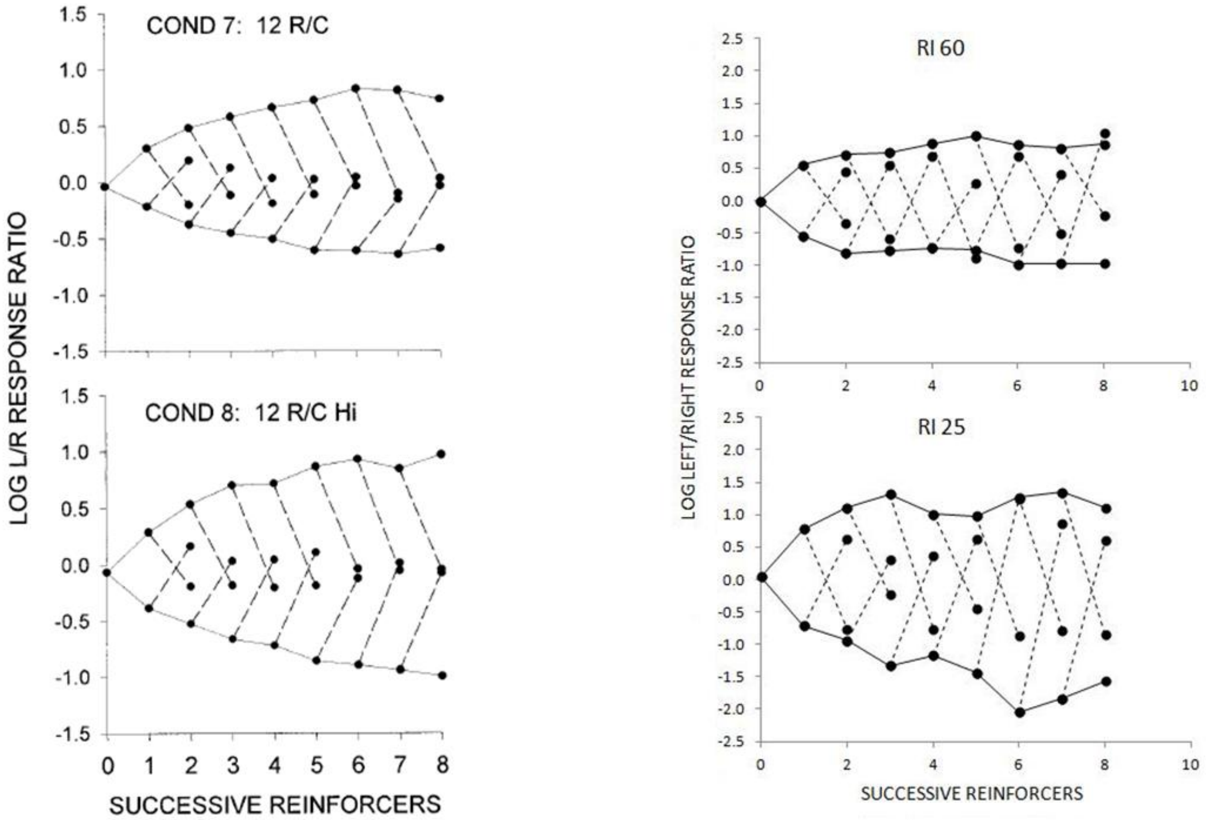

Figure 8. Log response ratios following selected sequences of left-right reinforcer deliveries for conditions with overall reinforcement rates of RI 25 and RI 60. Solid lines show *confirmations*, in which a reinforcer is delivered for a response on the same alternative as for the previous reinforcer delivery. Dotted lines show *disconfirmations,* which are reinforcer deliveries for a response on an alternative different from the one previously reinforced. On the left are two of the plots from Davison and Baum's (2000) paper (Figure 16 on p. 19).

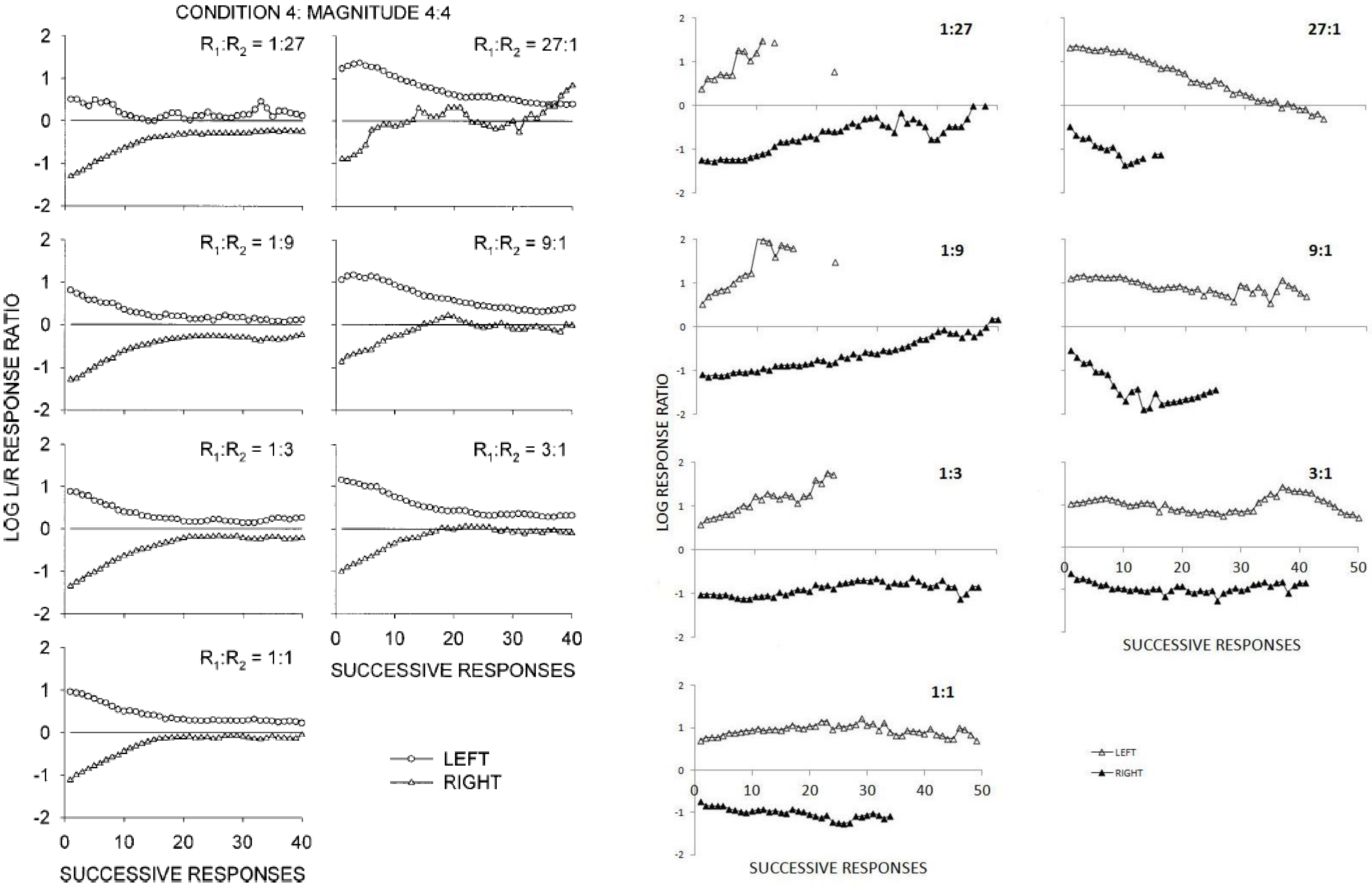

Figure 9. Preference pulses for the seven reinforcer ratios. Following reinforcement, preference was always for the alternative that delivered it. On the left are the plots from Davison and Baum (2003) paper (Figure 8 on p. 107).

Appendix A

Components of the Computational Model

Virtual Organism. The organism has a population of 100 possible behaviors. Each of the 100 behaviors is defined by an integer from 0 through 1023, thus there are 1024 behaviors that can appear in the repertoire, yet only 100 are present at any one moment. The 1024 possible behaviors are divided into four behavioral classes. The size of the repertoire, the range of behaviors, and the number of classes are arbitrary. All possible parameters of the model that can vary and are preset in a particular experimental configuration are listed in Table 1. From now on, a particular configuration commonly used in McDowell's lab is described.

To permit reproduction and mutation processes, individual behaviors in the organism's repertoire were also described by genotypes. The integer value of each behavior can be considered a phenotypic expression, which in turn can be written in binary form as a ten-digit string of 0s and 1s. This binary representation is denoted as that behavior's genotype. Ten-digit binary numbers range from 0000000000 to 1111111111, corresponding to the base-10 integers 0 to 1023. As noted by McDowell (2004), "one can view a behavior's genotype as consisting of a single digital chromosome that is made up of 10 genes (the ten characters in the string of 0s and 1s), each of which has two alleles (0 and 1)" (p. 300).

The possible range of integers was set to be circular, which means that it wrapped back upon itself from 1023 to 0. For this circular segment of integers, the distance between two integer phenotypes, x and y, is $| x - y |$ when going one way around the circle, and $1024 - | x - y |$ when going in the other direction. The *difference* between two integer phenotypes is defined as the smaller of these two distances. For example, the distance between 0 and 1023 is 1023 in the ascending direction, and is 1 in the descending direction. Thus, the difference between 0 and 1 is the same as the difference between 0 and 1023.

The behaviors in a population were sorted into one of four classes based on their phenotypes. Two target classes were defined by the 41 integers from 471 through 511, and the 41 integers from 512 through 552. The two target classes represented behaviors that produce reinforcement on the two alternatives of a concurrent schedule. Two extraneous classes of behavior were defined by the remaining 942 integers. At each time tick, the digital organism's population of 100 behaviors is repopulated in the process driven by the baseline probabilities of emissions for each of the four classes and the selection rules. Thus, each population of potential behaviors exists for one time tick, or generation, during which a behavior from one of the 4 classes is emitted. The probability of emission from each class is equal to the relative frequency of the behaviors in that class. For example, if a population consisted of 5 behaviors in the first target class, 10 in the second target class, and 85 in the extraneous class, then the probabilities of emission from the three classes are 0.05, 0.10, and 0.85. The fixed class structure of the population determines operant levels, or baseline probabilities of responding, for the classes. For experiments in this study, the probabilities were 0.04 (= 41/1024) for each of the target classes, and 0.46 (= 471/1024) for each of the extraneous class.

Fitness method. The concept of fitness is used for the definition of selection rules. A behavior that is selected by reinforcement is identified as "fit" with respect to environmental conditions. The other behaviors in the repertoire are considered more or less fit depending on how similar they are to the selected behavior. For example, if a target class contains 41 behaviors from 0 through 40, when an instance of that class is selected, a fitness criterion can be defined as the middle value of the class, which in this example is 20. The fitness of each behavior can be calculated as the absolute value of the difference between this criterion value and the integer that defines the behavior. Thus, smaller differences correspond to fitter individuals.

Parent Selection. According to the evolutionary principle, fitter behaviors should be more likely to be chosen for reproduction than less fit behaviors. A parental fitness function defines the relation between a behavior's fitness and its probability of being chosen as parent. The types of fitness functions that have been implemented in the model are listed in Table 1.

Reproduction. At each time tick, the repertoire is repopulated with a new set of child behaviors produced as the result of "mating" parent behaviors. The reproduction methods implemented in the model are listed in Table 1. For example, in *slicewise* reproduction the child genotype is created by combining a part of the mother's genotype and a part of the father's genotype.

Mutation Method. A subset of behaviors from the new population undergoes mutation, that is, their integer values change. Both rate and type of mutation can be varied. Various mutation methods that have been implemented in the model are shown in Table 1. For example, in *Gaussian* mutation, the original integer value of a "mutant" is designated as the mean of a Gaussian distribution of integer values with a specific standard deviation. The mutated value is then randomly selected from this distribution.

Appendix B

Computational Algorithm[2]

The following describes the genetic algorithm of the model using only the parameter settings that will be used in the proposed study. All other available parameter options and values are listed in Table 1, and in McDowell (2004).

If a behavior was emitted from one of the target classes and it was reinforced, then a midpoint fitness method (McDowell, 2004) was used to assign fitness values to each behavior in the population of potential behaviors. According to the midpoint fitness method, the fitness of a behavior is the *difference* between the behavior's phenotype and the phenotype at the midpoint of the target class. For example, if the just-reinforced behavior was emitted from the first target class, then the midpoint used to assign fitness values was 491. A behavior in the population with a phenotype of 400 would therefore have a fitness value of $| 400 – 491 | = 91$, and a behavior with a phenotype of 512 would have a fitness value of $| 512 – 491 | = 21$. Because the latter behavior is *less different* than the criterion midpoint, it is the fitter behavior. Thus, *smaller fitness values represent fitter behaviors.*

Once fitness values were assigned to the behaviors in the population, a linear parental selection function was used to select parents for mating on the basis of their fitness. This function expresses the probability density, *p(x)*, associated with a behavior of fitness, *x*, becoming a parent as

$$p(x) = -\frac{2}{9\mu^2}x + \frac{2}{3\mu}, \tag{1}$$

---

[2] This description of the computational model's algorithm is reproduced from McDowell (2008) , with the author's permission.

for $0 \leq x \leq 3\mu$, where μ is the mean of the density function. Notice that probability density decreases as fitness decreases (i.e., as the fitness value, x, increases) until it reaches a value of 3μ. Behaviors with fitness values of 3μ and greater have no chance of becoming parents. This same parental selection function was used in previous research (McDowell, 2004; McDowell & Caron, 2007; McDowell, Soto, Dallery & Kulubekova, 2006) and is the simplest linear density function that depends only on its mean. A father behavior was chosen from the population by drawing a fitness value at random from the distribution specified by Equation 1, and then searching the population for a behavior with that fitness. If none was found, then another fitness value was drawn from the distribution, and so on, until a father behavior was obtained. A mother behavior was chosen in the same way, but with the requirement that it be distinct from the father behavior. One hundred pairs of parents were obtained in this way. All parents were selected with replacement, which means that a behavior could be a parent more than once, and could have multiple partners. A detailed discussion of parental selection functions, including methods of drawing random values from them, can be found in McDowell (2004).

The process of assigning fitness values and selecting parents using Equation 1 occurred only if the emitted behavior was reinforced. If the emitted behavior came from one of the target classes but was not reinforced, or if it came from the extraneous class, then 100 pairs of parents were randomly selected with replacement from the population, with the requirement that the father and mother behaviors in a pair be distinct. Again, a given behavior could be a parent more than once and could have multiple partners.

A child behavior was created from each pair of parents by building a new 10-character bit string based on the parents' genotypes. Each bit in the child's string had a 0.5 probability of being identical to the bit in the same location of the father's bit string, and a 0.5 probability of being identical to the bit in the same location of the mother's bit string. This method of reproduction was used in previous research (McDowell, 2004; McDowell & Caron, 2007), where it was referred to as *bitwise* reproduction. It generates children that resemble their parents to varying degrees, where resemblance refers to the *difference,* as defined earlier, between the phenotypes of parents and offspring. The 100 child behaviors created in this way constituted the new population of potential behaviors.

A small amount of mutation was added to the new population by flipping one randomly selected bit in a percentage of behaviors chosen at random from the population. This method of mutation was referred to as *bitflip* mutation in previous research (McDowell, 2004; McDowell & Caron, 2007). The mutation rate specifies the probability that a behavior will mutate. For example, if the mutation rate is 1%, then each behavior in a population has a 0.01 probability of mutating. If a behavior does mutate, a location in the ten-character bit string is chosen at random and the bit at that location is changed from 0 to 1 or from 1 to 0. Using this method, a population of potential behaviors may have any number of mutants from 0 to 100, but across generations the mutation rate converges on the specified percentage.

Following mutation, the behaviors were sorted into classes and a behavior from one of the classes was emitted based on the relative frequencies of the behaviors in each class and then the process of selection, reproduction and mutation was repeated for the duration of the experiment. Each population constituted a generation, and lasted one time tick.

The target classes of behavior were defined so as to have maximum Hamming distances (Russell & Norvig, 2003) at their boundaries, which produced an effect similar to changeover delay (COD). The Hamming distance between two bit strings of equal length is the number of bits that must be changed to convert one string into the other. The Hamming distance between the upper boundary of the first target class (511 = “0111111111”) and the lower boundary of the second target class (512 = “1000000000”) was ten, which is the maximum Hamming distance for a ten-character string. The Hamming distance between the upper boundary of the second target class (552 = “1000101000”) and the lower boundary of the first target class (471 = “0111010111”) was also ten. This means that it was relatively difficult for recombination or mutation to cause a potential behavior to switch from one target class to the other. The result was responding that tended to be concentrated in bouts in a target class. Target classes with small Hamming distances at their boundaries tend to produce frequent switching between classes, just as often occurs in the absence of a COD in experiments with live organisms.