Learning Where (Not) To Cache: A Cognitive Model for Corvids

Elske van der Vaart (e.e.van.der.vaart@rug.nl)[1,2]
Charlotte Hemelrijk (c.k.hemelrijk@rug.nl)[1]
Rineke Verbrugge (l.c.verbrugge@rug.nl)[2]

[1]Theoretical Biology, University of Groningen, P.O. Box 14
9750 AA Haren, the Netherlands
[2]Artificial Intelligence, University of Groningen, P.O. Box 407
9700 AK Groningen, the Netherlands

Abstract

Computational modeling of cognition has provided many new insights into the human mind. In this paper, we use the same technique to advance our understanding of other, nonhuman minds: Those of corvids. This family of birds stores food under ground, saving it for later. This process of caching and recovery has been used to study many different aspects of corvid cognition, making it possible to build one integrated model that can be used to study many different cognitive phenomena. We start the construction of such a model by focusing on *memory*, and validate it by replicating three experiments by de Kort *et al.* Here, the caches of scrub jays are systematically stolen or moved, and the question is how this will affect their choice of cache sites. We use our model to reproduce the empirical data, and confirm its robustness by demonstrating that there are alternative outcomes that it *could not* have fit. In the process, we provide a new perspective on what exactly scrub jays may be learning and remembering.

Keywords: Computational model; cognition; corvid; scrub jay; *Aphelocoma californica*; caching; memory; learning.

Introduction

Computational models of cognition have significantly advanced our understanding of the human mind (Sun, 2008). Therefore, the same techniques hold promise for advancing our understanding of other, *nonhuman* minds. Essentially, what such models do, is to implement a theory of cognition as a working computer program, making it possible to very precisely test a theory's assumptions, conclusions and predictions. In the field of comparative cognition, many questions are being asked about animal memory, learning and problem solving (Premack, 2007). Innovative experiments have produced a wealth of empirical data, but results can be difficult to interpret. Computational models offer one way of testing the plausibility of competing explanations (Penn, Holyoak, & Povinelli, 2008).

Studies of jays, nutcrackers and other corvids - members of the extended family of crows - seem particularly likely to benefit from this approach. In terms of their cognitive abilities, corvids have been likened to apes: One species uses tools, another solves trap-tube problems, a third can infer hierarchical relationships (Emery & Clayton, 2004). But what makes corvids particularly attractive from a modeling perspective, is that some of the most interesting results have been obtained within a single experimental paradigm - that of caching and recovery.

Virtually all corvids cache: They bury food items under ground, saving them for later. It can be days, weeks, or months before they return to eat them. This behavior has been extensively studied in the laboratory, and depends on memory. Corvids' natural tendency to cache and recover has been used to study memory mechanisms, various kinds of learning, use of visual landmarks, future planning, and social cognition, among other things (de Kort *et al.*, 2006). Usually, in these experiments, the birds are presented with a discrete set of cache sites to choose from, a number of visual landmarks, the presence or absence of a conspecific, and very little else. This means a uniform setup is being used to study a diverse set of cognitive phenomena, making it possible to validate a single computational model across a large variety of experimental results.

In this paper, we start the design of such a model by focusing on *memory*, for cache and recovery events. To store these, we draw inspiration from the ACT-R cognitive architecture, and the way it uses *chunks* (Anderson, 2007). Here, a chunk is a declarative fact, stored in memory, with an *activation* that depends on its own history of use, as well that of related chunks. What this allows us to do, is to encode both a bird's *options* and its *memories* as chunks, and to have the most active option chunk determine its next action. Then, if we let memory chunks affect the activations of option chunks, we can simulate how previous cache and recovery events affect future cache and recovery choices.

We validate our cognitive model, and test its usefulness, by replicating three experiments by de Kort *et al.* (2007). This work features Western scrub jays, *Aphelocoma californica*, and looks at whether they learn to adjust their choice of cache sites in response to their experiences at recovery. In these experiments, a scrub jay in its cage is presented with a bowl of worms, and a number of visually distinct ice cube trays to cache in. After it has cached for fifteen minutes, all trays are taken away, only to be returned a day later, with all caches in one of the trays missing, or moved. This procedure is repeated across a number of trials, and the question is whether the scrub jay will change its caching behavior over time, and if it does, what the mechanism is. Is it responding to *punishment*, avoiding trays that are always pilfered, or is it responding to *reward*, preferring trays where it always finds its worms? Our computational model successfully reproduces the empirical data, and thereby gives a new answer to this question.

We show that *both* punishment and reward may be at work, and that learning is likely to involve recall of specific caching events. A systematic exploration of our model's parameter space confirms its robustness: First, there are plausible, alternative outcomes that it *cannot* generate, and second, there are no other initial assumptions that allow it to fit all three experiments. That is, with *only* punishment or with *only* reward, or with indiscriminate learning, our model *always* fails to match the birds' performance. In this way, our computational model provides a new perspective on what exactly scrub jays may be learning and remembering, and serves as a suitable starting point for more comprehensive architectures of corvid cognition.

Model

Our implementation of de Kort *et al.*'s (2007) experiments consists of two components: A *simulator* and a *cognitive model*. The simulator runs the experiments, while the cognitive model is a computational theory of the cognitive processes under concern. We assume that three factors account for the caching and recovery behavior of corvids: First, that corvids experience *inhibition* to return to recently visited sites; second, that they form *memories of caching events*, which influence where they will recover, and third, that they form *memories of recovery events*, which influence where they will cache. Motivational processes, that govern whether the birds want to cache or recover at all, are not considered; we simply assume that they want to cache in caching sessions and recover in recovery sessions.

The Basics of Chunks

At our model's core are two types of chunks: *Option* chunks and *memory* chunks. Option chunks represent the locations that are *available* for the bird to cache or recover in; memory chunks represent the *actual* cache or recovery events that the bird has experienced. Every chunk has at least two features, an *identifier* and an *activation*. A chunk's identifier specifies which individual tray section it represents, in which particular tray.

A chunk's activation A_i consists of three parts: *Base-level activation B_i*, *spreading activation S_i*, and *noise*; see Equation 1. A chunk's base-level activation B_i is computed according to Equation 2, following ACT-R's equation for base-level learning (Anderson, 2007). Here, t_j represents the elapsed time t since use j of chunk i, while d is a decay parameter. The weighing factor w_i is determined by chunk i's type, where the w_o of option chunks is 1, and the w_m of memory chunks is 2. This represents the idea that *actually* caching in a particular location is more memorable than simply *deciding* to cache there. The effect is that a chunk's base-level activation depends on its frequency and recency of use, plus its type. A chunk's spreading activation depends on the activation of other chunks, and will be discussed in the following sections. A chunk's noise value is re-computed every time it is evaluated, according to Equation 3, taken from ACT-R, where n is a parameter that we tune, and r is a random value between 0 and 1.

$$A_i = B_i + S_i + noise \qquad (1)$$

$$B_i = w_i \cdot \sum_j t_j^{-d} \qquad (2)$$

$$noise = n \cdot \ln(\frac{1-r}{r}) \qquad (3)$$

The Structure of Sessions

When our cognitive model is first initialized, its memory is empty, and the time is 0. Every subsequent caching or recovery event counts as one time step.

As soon as a caching session starts, the cognitive model encodes every individual section in every available tray as a *cache option chunk*. Every time it starts to cache, it computes the activation of all its cache option chunks, according to Equation 1, and selects the most active one; this causes the model bird to cache in the corresponding tray section, and counts as a use of that chunk. Once the cognitive model has selected its cache site, it caches there, and the corresponding *cache memory chunk* is given a use. Caching continues until the simulator asks the cognitive model to stop; this is determined by the average number of caches made by the real scrub jays in the same experiment.

A recovery session works in exactly the same way, except that it revolves around *recovery option chunks* and *recovery memory chunks*. Also, as we lack data on the number of recovery attempts made by the real jays, the simulator always prompts the model birds to try twenty recoveries, unless all caches are found before twenty attempts are made.

Inhibition of Return

To prevent the model birds from returning to recently visited sites, once a memory chunk is used within a session, it spreads *negative activation*, or *inhibition*, to the corresponding option chunk. See Equation 4 for recovery chunks, and Equation 6 for cache chunks. To work out the case of Equation 6: The higher the base-level activation B of the cache memory chunk cm_i, the lower the spreading activation S of the cache option chunk co_i, and the smaller the odds that the model bird will return to that cache site.

Knowing Where to Recover

What allows the cognitive model to relocate its caches, is the fact that every cache memory chunk spreads *positive activation* to the recovery option chunk that codes for the same location; see Equation 4. This has the effect that the cognitive model is *more* likely to try and recover in cups where it has actually cached items. Given the noise present in the system, it can of course make errors, however.

$$S_{ro_i} = B_{cm_i} - B_{rm_i} \qquad (4)$$

Learning Where To Cache

In de Kort *et al.*'s (2007) research, every experiment consists of multiple trials. Although every trial features new ice cube trays, with unique identifiers, the locations of trays are consistent across trials. A tray's *position* is what predicts whether or not its caches will be stolen or moved by the experimenter. Our model, then, considers all trays placed in the same location to be equivalent. It learns by taking the outcomes of previous recovery attempts into account: All recovery memory chunks of which the bird is *confident* spread activation to cache option chunks which code for the same or neighboring tray sections.

Confidence works as follows: Once the cognitive model has received the simulator's feedback concerning the outcome of a recovery attempt, it checks if it can remember actually caching in the current location. This process is not perfect; in trying to recall a cache memory chunk belonging to the exact tray section just probed, it can also accidentally retrieve a cache memory chunk for a section directly adjacent to it, either horizontally or vertically. Whichever of these eligible cache memory chunks has the highest activation will be selected for recall. Whether or not this is successful depends on ACT-R's probability of retrieval equation (Anderson, 2007); see Equation 5.

$$P_i = \frac{1}{1 + e^{\frac{-(A_i - \tau)}{n}}} \qquad (5)$$

Here, n is the same parameter that governs the noise production of Equation 3, and τ is a new parameter that governs the sharpness of the retrieval threshold. If a cache memory chunk can be retrieved, the new recovery memory chunk will have a confidence rating of 1; otherwise, it will have a confidence rating of 0. Once a chunk's confidence rating is set, a new experience will not affect it. Thus, a model bird can only learn from a recovery experience if it is confident that it has cached there; only then is the fact that it has relocated a worm, or failed to relocate a worm, really relevant to where it should cache in future.

What kind of activation a recovery memory chunk will spread, depends on its *success*; this is simply whether or not a simulated worm is found. Separate recovery memory chunks are kept for successful and unsuccessful recoveries. *Successful* recovery memory chunks spread *positive* activation; *unsuccessful* recovery memory chunks spread *negative* activation. This is summarized by Equation 6, where the expression $rm_{i,s,c}$ indicates a recovery memory chunk with identifier i, success s, and confidence level c.

$$S_{co_i} = -B_{cm_i} - \sum_{j \in T} B_{rm_{j,0,1}} + \sum_{j \in T} B_{rm_{j,1,1}} \qquad (6)$$

(where T represents the set of chunks that code for direct tray neighbors of chunk i)

Experiments

To validate our cognitive model, we test it against three experiments from de Kort *et al.* (2007). For each experiment, we average the outcomes of 1000 model runs, using the parameter values of Table 1. The choice of these values, and their effect on the model's performance, will be discussed in the section Parameter Search and Robustness.

Table 1: Parameter values used in the experiments.

	d, decay	*n, noise*	*τ, threshold*
experiments	0.10	0.35	3.00
range searched	0.05 – 0.5, by 0.05	0.05 – 0.5, by 0.05	1 – 3, by 0.25

Experiment 1 (Experiment 2b in de Kort *et al.* (2007))

In this experiment, the birds are presented with two ice cube trays, of which one is always replenished with fresh worms in the correct locations, while the other is pilfered and returned empty. Every caching trial, ten scrub jays may cache for fifteen minutes. When both trays are returned a day later, the birds may recover for ten minutes, finding one tray empty, and the other as they left it. This routine is repeated four times, with at least a day between trials, and unchanging locations for the *replenished* and *pilfered* trays.

Real Bird Results, de Kort *et al.* (2007) As can be seen in Figure 1A, as trials progress, the scrub jays cache proportionally more worms in the replenished tray. Statistically, de Kort *et al.* (2007) find a significant effect of trial on the number of caches in the replenished tray, and a significant difference between the two trays on trial four.

Computational Model Results Like the real birds, the model birds quickly learn to allocate proportionally more caches to the replenished tray. The learning curve of the model birds is similar to that of de Kort *et al.*'s (2007) scrub jays; see Figure 1A.
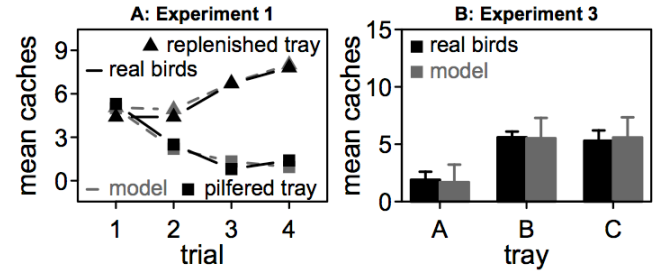

Figure 1: Results, real birds, de Kort *et al.* (2007) and computational model; 1A: Experiment 1, 1B: Experiment 3.

Experiment 2 (Experiment 4a in de Kort *et al.* (2007))

In this experiment, some caches are *moved* from one tray to another. Eight scrub jays are again presented with two trays, A and B, but now one of them, tray B, is unavailable for caching, as it is covered by a lid. A day later, during the recovery session, both trays are fully accessible. Now, the birds in the *control condition* find their caches returned to them in tray A, where they had left them; those in the *experimental condition* find their caches moved to tray B, with tray A now empty. After two such training trials, the birds are given a single test trial. It consists of one caching session, where both trays are fully accessible.

Real Bird Results, de Kort *et al.* (2007) On the test trial, the birds in the control group cache more in tray A than in tray B, while those in the experimental group seem to prefer tray B over tray A, though this difference is not significant (de Kort et al., 2007; see Figure 2A).

Computational Model Results As can be seen in Figure 2B, on the test trial, the model birds distribute their caches in a fashion that is qualitatively similar to that of de Kort *et al.*'s (2007) jays. The most notable difference is in the control condition, where the model birds show a slightly weaker preference for tray B than the real scrub jays.
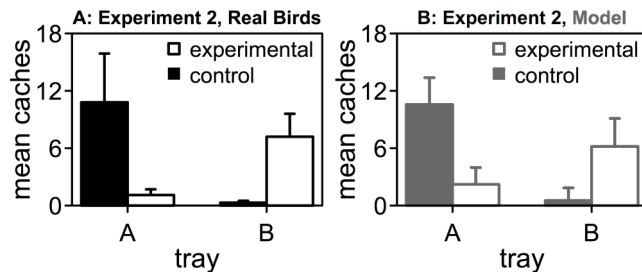

Figure 2: Results for Experiment 2; 2A: Real birds, de Kort *et al.* (2007), 2B: Computational model.

Experiment 3 (Experiment 4b in de Kort *et al.* (2007))

Like Experiment 2, this experiment involves movement of caches, but now using a third, neutral tray. It comprises two training trials, each consisting of a caching and a recovery session, and a single test trial, consisting of a caching session only. Every session, four birds are offered three trays, A, B and C. During the training trials, caching in trays B and C is not possible, as access to these trays is blocked. However, at recovery, the scrub jays find all their caches moved from tray A to tray B, leaving trays A and C empty. On the test trial, all three trays are available for caching.

Real Bird Results, de Kort *et al.* (2007) This experiment is replicated twice with the same subjects, with the outcome redrawn in Figure 1B. In the test trial, the scrub jays make significantly less caches in tray A as compared to trays B and C, while trays B and C are used equally often, statistically speaking (de Kort et al., 2007).

Computational Model Results Like the real scrub jays, in the test trial, our model birds cache approximately equally in trays B and C, and considerably less in tray A. A direct comparison can be found in Figure 1B.

Parameter Search and Robustness

All reported results are the product of the same parameter settings; see Table 1. These values were chosen after a comprehensive search for good fits, in all of the model's plausible parameter space. However, as Roberts and Pashler (2000) point out, the fact that a theory with free parameters is capable of reproducing empirical data is not in itself that informative. What is equally important, is to understand how strongly the model *predicts* the observed results, and if there are any plausible alternatives that it *cannot* reproduce. To determine this, we explore the model's performance in the *rest* of its plausible parameter space. In all, we evaluate 729 different combinations of *decay*, *noise* and *threshold*, by averaging the results of 100 model birds for Experiments 1, 2 and 3; see Table 1 for the range of values used.

First, we define as *qualitative fits* all results that match the empirical data in terms of the direction of tray preferences, with at least twenty percent more caches in the preferred trays. So, for Experiment 1, this requires more caches in the replenished tray than in the pilfered tray; for Experiment 2, more caches in tray A for the control group and more caches in tray B for the experimental group; and for Experiment 3, more caches in trays B and C than in tray A – all with differences of at least twenty percent. Using these criteria, we find a total of 565 qualitative fits. In other words, in about 78% of its parameter space, our model replicates the direction of observed tray preferences, if not their exact proportions. This raises the question: What does the model's behavior look like in the *rest* of its parameter space?

Taking a closer look, we find that the remaining 164 combinations of values have very high settings of *threshold* relative to *noise* and *decay*. What happens then, is that very few recoveries make an impression on the model birds, as they cannot specifically recall caching anywhere. As a consequence, they do not learn, or only very slowly. Instead, two kinds of behavior are possible: If *noise* is very low, the tendency to cache in familiar locations takes over, so that the model birds of Experiments 2 and 3 develop a preference for caching in tray A on test trials, the only tray that was also available during training. Otherwise, if *noise* is sufficiently high, this effect is suppressed, and the model birds cache equally in all trays available. This implies that, irrespective of its exact parameter settings, our cognitive model can only produce outcomes that reflect the learning displayed by the real birds, or that show no learning at all.

Discussion

In the previous section, we have demonstrated that our cognitive model can successfully reproduce the empirical results obtained by de Kort *et al.* (2007), and that this success is relatively robust with respect to the model's parameters. Three issues, however, remain to be discussed.

Implications of the Model for Real Scrub Jays

To assess what new insights our current work contributes, it is necessary to look at the conclusions drawn by de Kort *et al.* (2007), who were mainly interested in whether scrub jays learn by *punishment*, or by *reward*. In the end, they explain the birds' behavior in these experiments as inhibited caching in pilfered locations, using a 'memory at retrieval' process. The idea is that when the scrub jays attempt to recover a cache and find it missing, they recall the corresponding caching episode, and associate their current state of frustration with it, discouraging the birds from caching in that location again. As a result, the scrub jays learn to avoid trays that are pilfered, and place their caches in 'safe' trays instead. Thus, the mechanism at work is one of *punishment*.

This account is very close to what occurs in our cognitive model, except that our artificial birds respond to *both* punishment and reward. Every recovery leads to the creation of a recovery memory chunk; if the model bird is *confident* that it actually cached there, the recovery memory chunk will influence further cache choices. If it represents a success, it spreads *positive* activation to nearby cache option chunks; if it represents a failure, it spreads *negative* activation to nearby cache option chunks. Whether or not the artificial bird is *confident* of its recovery attempt depends on whether, at recovery, it can actually recall caching there. In other words, this is exactly de Kort *et al.*'s 'memory at retrieval' process, but with inhibited caching in pilfered trays and preferred caching in rewarded trays – punishment *and* reward. Our model uses both because it cannot fit the data otherwise. The main problem is with the control group of Experiment 2: On de Kort *et al.*'s (2007) test trials, the scrub jays in this condition cache significantly more in tray A (Figure 2A), where they have previously cached and recovered. If we run our cognitive model with spread of negative activation only, and the parameters of Table 1, we get the results of Figure 3A: The model birds of the control condition cache equally in trays A and B. As they experience few unsuccessful recoveries, they do not learn to prefer one tray over another. In fact, with *just* punishment, or with *just* reward, we find no qualitative fits of the empirical data at all, anywhere in the model's plausible parameter space.
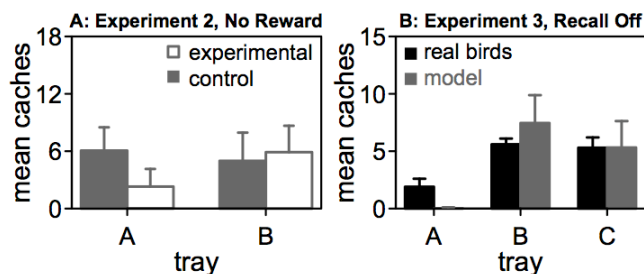

Figure 3: Results of altered models; 3A: Experiment 2, computational model, 3B: Experiment 3, real birds, de Kort *et al.* (2007) and computational model.

This effect might have been predicted without running the simulation. What the model contributes, is that it allows us to easily integrate different explanations. The idea that successful recoveries reinforce particular caching strategies is not new; Clayton *et al.* (2005) already propose it. However, de Kort *et al.* (2007) discard this explanation because it does not account for the results of Experiment 3. Our model shows that punishment and reward are not mutually exclusive, and that, in fact, only the two mechanisms *together* are capable of successfully reproducing all of the empirical data.

We also draw a second conclusion from our work, which is that recall of specific caching events really is necessary to produce the results obtained by de Kort *et al.* (2007). In the first version of our model, our artificial birds were automatically impressed by *every* recovery attempt. In Experiment 3, this creates unforeseen effects. If we assume that every failure to recover a worm inhibits caching in that tray, then the model birds of Experiment 3 always learn to prefer tray B over tray C, *even* if we only allow learning through punishment. This is illustrated in Figure 3B, using the parameter values of Table 1. At first sight, the result is counterintuitive: The artificial birds have only lost caches from tray A, and they only learn from negative experiences, so why do they differentiate between trays B and C?

The answer is that, during recovery, once the model birds give up on tray A, they start probing the other trays, and invariably experience fewer unsuccessful recoveries in tray B, because their own caches have been moved there, while tray C is empty. Only if recovery attempts are linked to specific caching events, can the birds learn to avoid caching in tray A but treat trays B and C equally. For us, this insight came only after explicitly simulating the experiment, illustrating the contribution that models can make to clarifying what specific theories and assumptions predict.

Validation of the Model's Recovery Behavior

As de Kort *et al.*'s (2007) experiments focus on caching strategies, recovery behavior is not reported. Therefore, we have not fit this aspect of our cognitive model to the empirical data. Other publications, however, offer clues that allow us to judge its performance. One issue concerns whether the real scrub jays in Experiments 2 and 3 learn to look for worms in tray B, where their caches are moved to, rather than in tray A, where they left them. Our simulated birds, in any case, cannot learn to shift their attention at recovery. Work by Watanabe (2005) suggests that this lack of learning is perhaps not too much of a simplification; there, in a similar setup, scrub jays find their entire *tray* of caches moved at recovery, with a different tray left in its place. These birds, in other words, have the tray's visual identifiers to guide them to the new locations of their caches. But even they do not learn to prefer the displaced tray over the course of three trials. Another issue concerns the scrub jays' basic recovery accuracy: For this, we turn to Clayton and Dickinson (1999), who test scrub jays in similar setups, though involving shorter retention intervals.

They report that the birds direct 76 ± 5% and 81± 5% of their recovery attempts to cache sites. When we calculate the same statistic for our model birds in the control condition of Experiment 2, using the parameters of Table 1, we find an accuracy of 83 ± 18%, which seems to be in the right range. We hope to further validate our model's recovery behavior in future.

Plausibility of the Model's Implementation

A number of core aspects of our model are derived from the cognitive architecture ACT-R (Anderson, 2007), and in particular, its declarative memory module. But ACT-R is a cognitive architecture designed for humans, and our use of it for birds might seem problematic. However, we do not think that this is the case: Although built for humans, ACT-R's declarative memory module captures fundamental properties, such as recency and frequency effects, that seem to apply to all species. In addition, previous research has used the full architecture to model experiments with mice (Belavkin, 2001) as well as monkeys (Wood, Leong, & Bryson, 2004), so there is some precedence for our approach.

More fundamental, perhaps, are the changes we make to the architecture itself. While ACT-R allows for spreading activation, it is a fixed amount, which spreads between related chunks, depending on the strength of the association between them. In our cognitive model, it is a chunk's *own* activation that spreads, and this activation can even be negative, inhibiting a chunk's retrieval. The main function of this mechanism is to prevent the model birds from continuously caching or recovering in the same location. It is possible that the same effect could be achieved by instead *increasing* the activation of all other chunks, but our solution is computationally easier, and seems to be acceptable in the model's current state of development. Interestingly, other recent ACT-R adaptations also make similar changes to the architecture: Van Maanen & van Rijn (2007) let activation spread between chunks of different types, and Juvina & Taatgen (2009) also attach negative activation to chunks to suppress their retrieval.

Conclusions and Future Work

In this paper, we have presented a computational model of corvid cache and recovery cognition, and have shown that it can successfully replicate three experiments by de Kort *et al.* (2007), with robust performance. In this work, scrub jays must learn where it is safe to cache. Our work brings the novel insight that *both* positive and negative reinforcement are likely to be involved. In future work, we aim to apply the same cognitive model to other cache and recovery experiments, in order to study new cognitive phenomena.

Acknowledgments

References

Anderson, J. R. (2007). *How can the human mind occur in the physical universe?* Oxford: Oxford University Press.

Belavkin, R. (2001). The role of emotion in problem solving. *Proceedings of the AISB'04 Symposium on Emotion, Cognition and Affective Computing* (pp. 49-57). Heslington, York, England.

Clayton, N. S., Dally, J. M., Gilbert, J., & Dickinson, A. (2005). Food caching by western scrub-jays (*Aphelocoma californica*) is sensitive to the conditions at recovery. *Journal of Experimental Psychology: Animal Behavior Processes, 31*(2), 115-124.

Clayton, N. S., & Dickinson, A. (1999). Memory for the content of caches by scrub jays (*Aphelocoma coerulescens*). *Journal of Experimental Psychology: Animal Behavior Processes, 25*(1), 82-91.

de Kort, S. R., Correia, S. P. C., Alexis, D. M., Dickinson, A., & Clayton, N. S. (2007). The control of food-caching behavior by western scrub-jays (*Aphelocoma californica*). *Journal of Experimental Psychology: Animal Behavior Processes, 33*(4), 361-370.

de Kort, S. R., Tebbich, S., Dally, J. M., Emery, N. J., & Clayton, N. S. (2006). The comparative cognition of caching. In E. A. Wasserman & T. R. Zentall (Eds.), *Comparative cognition: Experimental explanations of animal intelligence*. Oxford: Oxford University Press.

Emery, N. J., & Clayton, N. S. (2004). The mentality of crows: Convergent evolution of intelligence in corvids and apes. *Science, 306*, 1903-1907.

Juvina, I., & Taatgen, N. A. (2009). A repetition-suppression account of between-trial effects in a modified stroop paradigm. *Acta Psychologica, 131*(1), 72-84.

Penn, D. C., Holyoak, K. J., & Povinelli, D. J. (2008). Darwin's mistake: Explaining the discontinuity between human and nonhuman minds. *Behavioral and Brain Sciences*(31), 109-178.

Premack, D. (2007). Human and animal cognition: Continuity and discontinuity. *Proceedings of the National Academy of Sciences of the USA, 104*(35), 13861-13867.

Roberts, S., & Pashler, H. (2000). How persuasive is a good fit? A comment on theory testing. *Psychological Review, 107*(2), 358-267.

Sun, R. (Ed.). (2008). *The Cambridge handbook of computational psychology*. Cambridge, NY: Cambridge University Press.

van Maanen, L., & van Rijn, H. (2007). An accumulator model of semantic interference. *Cognitive Systems Research, 8*(3), 174-181.

Watanabe, S. (2005). Strategies of spatial learning for food storing in scrub jays. *Journal of Ethology*(23), 181-187.

Wood, M. A., Leong, J. C. S., & Bryson, J. J. (2004). ACT-R is almost a model of primate task learning: Experiments in modelling transitive inference. *Proceedings of the 26th Annual Meeting of the Cognitive Science Society* (pp. 1470-1475). Chicago: Lawrence Erlbaum Associates.