Re-Encoding and Human Memory

Nathaniel Neligh*

August 18, 2023

Abstract

In this paper we describe and characterize an informational tool which is primarily useful for improving the performance of noisy memory systems: re-encoding. Re-encoding is the process of periodically attempting to restore damaged or decayed memories to a better state. This can help avoid the accumulation of errors which ultimately leads to total information loss. We also show that a Bayesian form of re-encoding in the human mind could be responsible for a number of features of human memory including constructed memory, the poor correlation between memory accuracy and confidence, rehearsal effects, and specific spacing effects. It can also explain the observed way that memories decay over time.

1 Introduction

In this paper we have three primary goals: (1) to introduce the idea of re-encoding information for improved transmission and memory storage, (2) to show that re-encoding can explain several well known features of human memory, and (3) to provide some baseline theoretical results regarding re-encoding. The human memory provides a good introduction to the topic, so I begin by discussing application.

Human memory displays a wide array of quirks and tendencies that are difficult to explain in a well optimized memory system. In this paper we examine five well documented memory phenomena which currently lack a unifying theoretical explanation. One such phenomenon is constructed memory, a phenomenon where the brain creates false memories to fill in gaps or to fit better with other available information.[1] Relatedly, there is generally poor correlation between memory accuracy and

*University of Tennessee, Knoxville

[1] Schacter (2001); Stark et al. (2010)

the confidence people have in their memories.[2] Another quirk is the retrieval or testing effect in which being asked to explicitly recall a piece of information increases a subjects later ability to recall that information.[3] The spaced recall effect builds on the retrieval effect and tells us the greatest performance improvement will be seen if the retrievals are spaced evenly in time.[4] Finally, there is the empirical shape of the function relating memory performance to time.[5]

In this paper we propose a novel mechanism which explains all five of these previously disparate effects: Bayesian re-encoding. Bayesian re-encoding is a process where an agent examines the current, potentially damaged, state of a memory and then restores that memory to what they believe was most likely its original state. Consider a piece of paper with text "Econ_mics" where the blank represents a letter that is smudged to the point of illegibility. Using knowledge of the language and context, one can conclude the the word was probably originally "Economics". Knowing this, one can erase the word and rewrite "Economics", re-encoding the information and resetting the memory to a pristine un-decayed state. The newly re-encoded word is easier to read and more resilient to possible future smudges. This is the idea behind our proposed Bayesian re-encoding model. To our knowledge this type of active Bayesian re-encoding has not been previously considered. The closest area of the literature would be the use of error correction codes in computing, but that approach is very local and can only be applied to very limited memory errors (Chen and Hsiao, 1984).

How does Byaesian memory re-encoding explain the memory feature of interest? The following paragraphs discuss how re-encoding can explain overconfidence and constructed memory; rehearsal and spacing effects; and the rate of memory decay.

To begin, we explain the lack of relation between memory accuracy and memory quality. As we show, the re-encoding process destroys information about how accurate a memory was before it was re-encoded. The information about how decayed a memory was before is wiped away during the Bayesian re-encoding process.[6] This explains the disconnect between confidence and memory accuracy. If we combine this loss of quality information with the Bayesian way new evidence is incorporated into beliefs during a re-encoding we can also explain constructed memory.

Given the loss of quality information, why use re-encoding? Re-encoding a memory does improve memory performance by resetting the accumulation errors and this benefit can explain the benefits

[2] Simons et al. (2010); Chua et al. (2004); and Leippe (1980)

[3] Brewer et al. (2010)

[4] Mulligan and Peterson (2014); Jacoby (1978)

[5] Ebbinghaus (1885); Averell and Heathcote (2011)

[6] This information can be stored elsewhere but it is no longer present in the original data. Using extra memory space to store information about data accuracy is generally less useful than devoting that space to storing the original data with greater redundancy.

of explicit recall. Given that recollection could logically trigger a re-encoding, adding a re-encoding should improve memory performance. Further, in cases with multiple re-encodings, the benefit is greatest when they are evenly spaced, consistent with the observed rehearsal spacing effect.[7]

Empirical observations show that memories have a constant or decreasing hazard rate.[8] A constant rate of forgetting contradicts the way errors accumulate in most memory systems. However, re-encoding resets memory decay patterns in a way that keeps the hazard rate constant and approximates power law decay well.

While explanations have been offered for most of the described memory features other than the power law of decay, most of these explanations do not arise from an optimizing Bayesian framework with the exception of some explanations for constructed memory.[9] Further, no existing theory provides a unified explanation for all of these phenomena.

Contributing to the realism of our concept, there is also some neurological evidence which is consistent with re-encoding in the human brain. Buhry et al. (2011) find that the brain essentially replays memories during memory formation and maintenance. This could easily produce a re-encoding effect.

Section 2 establishes a novel model of memory systems. Section 3 provides theoretical results showing how the re-encoding process destroys information. Section 3.2 shows that re–encoding will be beneficial if the memory performance function is locally concave over short time frames. In section 3.3, we show that regular re-encoding leads to consistent forgetting rates which strongly resemble those that have been observed experimentally.[10]

In Section 4, we generalize several results from the memory discussion both theoretically and conceptually by dropping a major symmetry assumption and moving to a more abstract setting. Memory behaves like a sequence of noisy information channels. With imperfect memory, a person is playing a game of telephone with their future self. By correcting errors as they go, the player can do better than by letting them accumulate between channels. However, memory is not the only environment which has this structure, Many telecommunications and social environments involve passing information through multiple agents in sequence.

Finally, in Section 5 we consider non-Bayesian re-encoding schemes. In this chapter we offer a number of results which can make it easier to find the utilitarian optimal re-encoding, although in

[7] See Mulligan and Peterson, 2014 and Jacoby, 1978. Note that this is somewhat different than the most commonly studied spacing effect where spaced exposures to stimuli generate more persistent memories. For an example of that spacing effect see Leicht and Overton (1987).

[8] Ebbinghaus (1885); Averell and Heathcote (2011)

[9] The exception is constructed memory where a Bayesian explanation has been offered by Hemmer and Steyvers (2009)

[10] Murre and Dros (2015); Averell and Heathcote (2011)

general this problem remains difficult.

2 Model Environment

Consider an environment where an agent wishes to remember a state of the world. They learn the state of the world and set it into memory at time $t = 1$ and use it later at $t = \tau$.

Formally, there is a state of the world $\theta \in \Theta$. The state of the world is stored in a memory system. Conceptually, a memory system could be a hard drive, a brain, or a piece of paper with a password written on it. Any physical system that encodes information can act as a memory system. The memory system can is always in some memory state $m \in \mathcal{M}$. For example, if the memory system is made up of 4 bits, one memory state would be $(1, 0, 1, 0)$. We assume that $|\mathcal{M}| \geq |\Theta|$ and that $|\mathcal{M}|$ and $|\Theta|$ are both finite so $\Theta = \{\theta_1, \theta_2, ..., \theta_{|\Theta|}\}$ and $\mathcal{M} = \{m_1, m_2, ..., m_{|\mathcal{M}|}\}$. This finite approach simplifies notation dramatically.

A memory system generally has three parts: an encoding which moves information into the memory medium, storage which holds the information over time, and retrieval which extracts the information from the memory medium when it is needed.

In our model, the encoding is a mapping from Θ to $\mathcal{M}$. We call the range of this mapping the set of *initial memory states* or the set of *encoding states*. We denote the memory state which maps from θ as $\tilde{m}_\theta$ and the set of all encoding states as $\tilde{\mathcal{M}} \subset \mathcal{M}$. We do not require that the encoding be chosen optimally, although we will be using optimal encodings in the examples. For future convenience we define the $|\Theta| \times |\mathcal{M}|$ encoding matrix $N[i, j] = 1$ if $\tilde{m}_{\theta_j} = m_i$ and 0 otherwise. We assume that N is deterministic, as there is no benefit to stochastic encoding.

The storage part of the model examines what happens to the memory state over time. Say there is a $|\mathcal{M}| \times |\mathcal{M}|$ Markov transition matrix D where $D[i, j]$ is the probability of memory state m_j transitioning to memory state m_i in one period. Let $\alpha \in \Delta\mathcal{M}$ be a distribution over memory states represented as a vector. The matrix D can be used as a transformation on such distributions $D : \Delta\mathcal{M} \to \Delta\mathcal{M}$ where $D\alpha$ represents the distribution over memory states after a period of decay.

The probability of being in memory state m_i given initial memory state $\tilde{m}_\theta = m_j$ after t periods is given by

$$D^t[i, j]$$

Note, we use brackets to indicate elements of matrices and vectors because we will be using powers and subscripts of matrices, making standard notation confusing.

The retrieval part of the model is mapping from retrieval time and memory state to posterior over states of the world. We require this be done using Bayes Theorem, so the posterior probability of the initial state θ given memory state m at time t

$$\gamma(\theta_i|m_j,t) = \frac{\left(D^t N \boldsymbol{e}_i\right)[j]\pi[i]}{\sum_{k=1}^{|\Theta|}(N D^t \boldsymbol{e}_k)[j]\pi[k]}$$

Where $\pi(\theta)$ is the prior probability of state θ and $\boldsymbol{e}_i$ is a vector with a 1 in place i and a 0 everywhere else.

Because we assume that the retrieval is done in a Bayesian way, we can define a memory system through D and N.

At time τ, after updating their belief based on the memory state, the player will be given the opportunity to pick an action $a \in \Theta$. We assume a matching-based utility.

$$u(a,\theta) = \mathbf{1}(a = \theta)$$

This utility function allows us to avoid considering trade-offs between memory precision and resilience, and lets us focus on re-encoding.

This provides an optimal action picking strategy $a(m,t)$ for the agent.

$$a\left(m_i,t\right) = \arg\max_j \left(D^t N \boldsymbol{e}_j\right)[i]\pi[j]$$

Note that this gives the index of the optimal action. We can represent this as an $|\mathcal{M}| \times |\Theta|$ action matrix A_t where $A_t[i,j] = 1$ if $a\left(m_j,t\right) = i$ and 0 otherwise

Given $a(m,t)$ and final distribution of memory states $D^\tau N \pi$, we can derive the raw state dependent performance function

$$\rho(\tau) = \sum_i p\left(a(m,t) = \theta_i | D^\tau N e_i\right)\pi(\theta) = Tr(A D^\tau N \Pi) \tag{1}$$

Which reflects the probability of the agent correctly remembering θ after τ periods. Here Π is a diagonal matrix with π along the diagonal.

Remark 1. The principal of Blackwell (1953) dominance and garblings guarantees that $\rho(\tau)$ is weakly

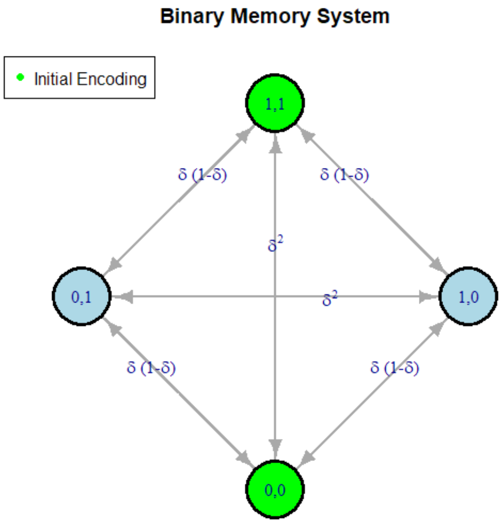

Figure 1: Binary memory system. $n = 2$ and $|\Theta| = 2$

decreasing.

In addition, by construction, $\rho(\tau) \in [0, 1]$. Having outlined the general definition and properties of a memory system and performance function, we present a few examples.

2.1 Binary Memory System Example

First we consider a relatively standard memory system based on bits.

Take a binary memory system with n bits. There are two states of the world and therefore two initial encoding states $(1, 1, 1, \ldots$ and $0, 0, 0, \ldots)$. Memory states decay through bit flips, and every bit flips probability δ each period.

This memory system can be represented by the network picture in Figure 1.

Assuming n is odd, the performance function in this case is given by

$$\rho(t) = 1 - BinomCDF\left(\tfrac{n-1}{2}, b(t), n\right)$$

Where $b(t) = 0.5 + 0.5(1 - 2\delta)^t$ is the probability of a bit being in the correct position after t periods. This is essentially saying that the answer will be correct as long as the majority of bits are in the correct position. Here $b(t)$ gives the probability of a bit being in the correct position after t

periods.

2.2 Re-encoding

Having established what a memory system is, we move on to explaining re-encoding. Re-encoding is essentially a way of resetting the memory decay process.

Definition 1. A re-encoding is a mapping between memory states which is applied to the memory state between two given Markov decay periods. We represent a re-encoding as a stochastic matrix R where $R[i, j]$ is the probability of switching memory state m_j to memory state m_i during the re-encoding.

2.2.1 Bayesian Re-encoding

In general re-encodings can take many forms, but we will focus on a specific, intuitive type of re-encoding.

Definition 2. If a Bayesian re-encoding occurs at time t, then $m_{t+1} = \tilde{m}_{\hat{\theta}}$, where $\hat{\theta} = \arg\max_{\theta} \gamma(\theta|m_t, t)$.

Here m_t is the memory state at time t. In other words re-encoding means finding the most likely initial state given the current memory state and then setting the memory system to the corresponding encoding state. Note that if $|\mathcal{M}| = |\Theta|$ then it is possible for every memory state to be an encoding state. In this case, re-encoding would do nothing. This is why we assume $|\mathcal{M}| \geq |\Theta|$ in our discussion.

In the binary case, re-encoding will revert the memory state to the initial memory state that is most similar to the current memory state in terms of single symbol mutations. For example, if the initial memory states are $(1, 1, 1)$ and $(0, 0, 0)$ and the current memory state is $(1, 1, 0)$, then re-encoding will revert to $(1, 1, 1)$.

We focus specifically on Bayesian re-encoding within the broader scope of re-encodings because it is easy to work with, performs well when N is well chosen, and because it makes intuitive sense. It is also optimal in some environments, but we will leave the question of optimal re-encoding until Section 5.

A re-encoding scheme or plan is a sequence of time periods with re-encoding occurring after each listed period.

2.3 Absorbing Star Memory System Example

Now we consider a novel memory system to give another example of Bayesian re-encoding.

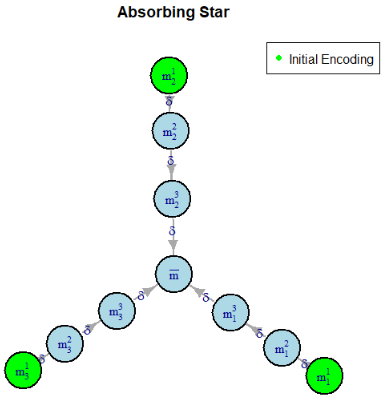

Figure 2: Absorbing star memory system. $k = 3$ and $|\Theta| = 3$

This structure has $k * |\Theta| + 1$ states. There is one absorbing $\bar{m}$ state in the middle and then for each θ there are k states including the initial encoding state $\tilde{m}_\theta = m_\theta^1$ and a sequence m_θ^i for $i = \{2, 3, ..., k\}$. Each period, if the memory state is m_θ^i for $i < k$, then the memory state transitions to m_θ^{i+1} with probability δ and remains the same with probability $1 - \delta$. If the current memory state is m_θ^k then it transitions to $\bar{m}$ with probability δ and remains the same with probability $1 - \delta$. See Figure2 for a visualization of the transitions

The performance function for the absorbing star is

$$\rho(t) = \frac{1}{|\Theta|} + \left(\frac{|\Theta|-1}{|\Theta|}\right) BinomCDF(k, \delta, t)$$

In the absorbing star, re-encoding would place the memory state back at the tip of whichever arm it was currently on. For example, m_3^3 would be reverted to m_3^1. If the current state was $\bar{m}$, a random action would be selected.

2.4 Simplex-Like Memory Structures

To show results in a simple and easy to interpret manner, we impose some additional structure on the setting. First we define a symmetry condition.

Let $P(m, m', D)$ be the set of transition probability sequences along all paths between m and m' based on the stochastic transition matrix D. Example if one path between m and m' includes a step with probability 0.5 and another step with probability 0.23, and no other steps, then $P(m, m', D)$

would include the sequence (0.5,0.23) as an element. Note steps where the memory state does not change must still be included in the sequence. Intuitively, $P(m, m', D)$ is the position of m' relative to m as defined by possible paths between the two.

Define $\xi(m)$ as an ordered list of $|\Theta|$ sets of probability sequences such that $\xi(m)[i] = P\left(\tilde{m}_{\theta_i}, m, D\right) \forall i$. Essentially, $\xi(m)$ can be thought of as coordinates within the network relative to the encoding states.

Definition 3. A memory system is **Simplex-Like** if it satisfies the following. Take any ordered list of $|\Theta|$ sets of probability sequences $\tilde{\xi}$and $\tilde{\xi}'$ a permutation of $\tilde{\xi}$. Let $K(\tilde{\xi})$ be the number of memory states m such $\tilde{\xi} = \xi(m)$. It must be that $K(\tilde{\xi})$=K($\tilde{\xi}'$).

We call this Simplex-Like because it guarantees that each encoding memory state is essentially "equidistant" from every other encoding state where distance is taken along every possible path. The graphs of such memory systems can be fitted symmetrically in simplexes (of dimension greater than 1) with the encoding states as end points. The $K(\xi)$ gives the count of memory states in the same relative position. The condition requires that for any group of memory states in some "position" relative to each encoding state in some order there must be an equally large set of memory states in the equivalent position relative to the encoding states in every other order.

One immediate effect of simplex-like memory systems when combined with uniform π is that the player is equally likely to confuse any pair of states. Formally, $p\left(a = \theta_i | \theta_j\right) = p\left(a = \theta_k | \theta_l\right) \forall \theta_i \neq \theta_j, \theta_k \neq \theta_l$ because for every memory state leading θ_i to be chosen when the true state was θ_j there is an equally probable memory state leading θ_k to be chosen when the true state was θ_l.

Simplex-Like is a strong symmetry condition, but it applies to a significant number of interesting memory systems. Both the two-state binary example and the absorbing star example previously introduced are simplex-like. More generally, any absorbing star memory system is simplex-like. Binary memory systems with arbitrary numbers of states are often approximately simplex-like when there is independent bit-wise decay and encodings are optimal, but there are sometimes small deviations for technical or rounding reasons.[11] This near simplex-like tendency comes from the fact that it is generally better to space out encoding states as much as possible to avoid confusion. For example, in the case with case with $|\Theta| = 3$, independent bit flip errors, and three bits, an optimal set of initial encodings would be $(1, 1, 1)$, $(0, 0, 1)$, $(1, 0, 0)$. This creates a simplex-like memory setting, because each initial encoding has two differences from each other initial encoding.

To illustrate how the definition of Simplex-Like works in practice, we provide a simple example of

[11] Lin et al. (2018)

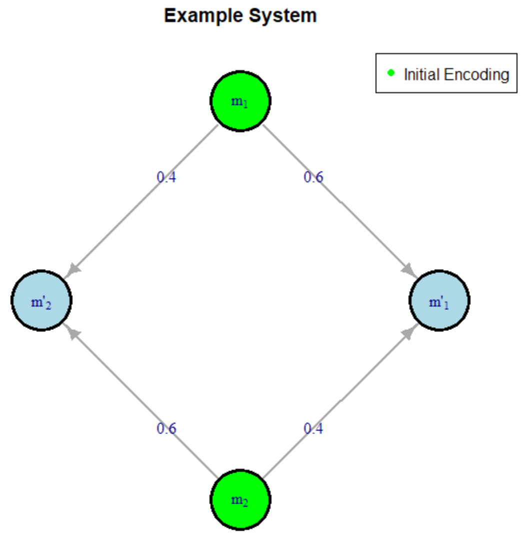

Figure 3: A simple memory system to illustrate the definition of Simplex-Like. $|\Theta| = 2$

a non-trivial simplex-like memory system.

2.5 Basic Simplex-Like Example

Consider a four state memory system with $|\Theta| = 2$. There are two encoding states, m_1 and m_2, and two non-encoding states m'_1 and m'_2. Memory state m_1 transitions to m'_1 with a probability of 0.6 and transitions to m'_2 with a probability of 0.4. Memory state m_2 transitions to m'_2 with a probability of 0.6 and transitions to m'_1 with a probability of 0.4. There are no other transitions.[12] Figure 3 shows this memory system visually.

This memory system contains no cycles which makes notation easier, since there are no infinite paths and the number of paths is itself finite. We begin by considering the non-encoding states. There is one path leading from m_1 to m'_1. It has one step with a probability of 0.6. There is also one path from m_2 to m'_1 which consists of a single step of probability 0.4. Therefore $\xi(m_1) = [\{[0.6]\}, \{[0.4]\}]$. The first element of the list is the set of paths from m_1 to m'_1, which has only one element of length one. The second element is the set of paths from m_2 to m'_1 which again has only one element of length one. Conversely $\xi(m_2) = [\{[0.4]\}, \{[0.6]\}]$. The first element of the list is the set of paths from m_1 to

[12] Note that the encoding states are not chosen optimally in this example.

m_2', and the second element is the set of paths from m_2 to m_2'.

No paths lead from any encoding state to any encoding state (in fact no paths lead to any encoding state at all). Therefore $\xi(m_1) = \xi(m_2) = [\phi, \phi]$. We now consider the next part of the definition. The list $[\{[0.6]\}, \{[0.4]\}]$ has only one permutation, $[\{[0.4]\}, \{[0.6]\}]$. As we have shown $K([\{[0.6]\}, \{[0.4]\}]) = L([\{[0.6]\}, \{[0.4]\}]) = 1$ so that part of the definition is satisfied. In addition $[\phi, \phi]$ either has no permutation or is its own permutation, so $K([\phi, \phi]) = 2$ satisfying the definition trivially. $K(\bullet) = 0$ for any ξ not mentioned since we have already accounted for $\xi(m)$ for all m in the system. Therefore, the definition is satisfied.

3 Psychological Results

Having fully established our setup, we now provide the results which explain the observed quirks of human memory. For this section we assume memory systems are simplex-like and that the prior π is uniform.

3.1 Constructed Memory and Confidence

We begin by talking about constructed memory and the connection between confidence and memory quality. Both of these results come about as the result of the following result:

Proposition 1. *Given, simplex like memory, uniform prior, and matching-based utility, the probability of selecting action action, a, t periods after a bayesian re-encoding depends only on the immediately previous Bayesian re-encoding and t.*

For proof see Appendix A.1.

3.1.1 Constructed Memory

A constructed memory is a false but seemingly real memory which is often, but not always, created by the brain to fill a gap or match with other evidence.[13] For example, if a person's parents regularly talk about a time when the Christmas ham got lit on fire, that person might develop a vivid memory of the event even if they were too young to form long term memories or they weren't there. Along a similar vein, Roediger and McDermott (1995) show that an experimenter can create false memories

[13] Schacter (2001); Stark et al. (2010)

of a word in a list by including many associated words. The brain notices the pattern and fills in the blanks.

Proposition 1 tells us that conditional action probabilities depend only on the previously encoded state of the world. Once a false state of the world has been encoded, it will have the same persistence as a true state of the world.

For example, say a coin flip is encoded as $(1, 1, 1, ...)$ for heads or $(0, 0, 0, ...)$ for tails. Bits flip with some fixed probability. The true state was tails, encoded originally as $(0, 0, 0, ...)$. After some time, the memory is in a state with a near even number of 1s and 0s with slightly more 1s. The player would re-encode their memory to $(1, 1, 1, ...)$ essentially constructing a memory of heads.

Constructed memory could also happen in the model in response to outside information. Consider a similar situations with a coin flip encoded in bits. The true state was tails, but after some time the flipping of bits leads to an uninformative memory state with an identical number of 1s and 0s. Say that the player receives some informative exogenous signals indicating that the coin flip was heads. Now heads is the posterior mode, and the memory is re-encoded as $(1, 1, 1, ...)$. Someone telling the player the coin flip came up heads could cause the player to construct a memory of heads.

The concept of a Bayesian reconstruction source for false or distorted memory has been considered by psychologists Hemmer and Steyvers (2009). Their experiment found evidence which was in line with the Bayesian reconstruction hypothesis.

3.1.2 Confidence

In the constructed memory example, we see that re-encoding effectively deletes all of the information a memory state contains about its quality. Consider the two state binary encoding environment. A highly informative memory state will have a much higher proportions of 1 or 0s, while an uninformative memory will have a similar number of each. During a re-encoding step, a memory state with just one more 1 will be re-encoded as all 1s. A very weak signal becomes indistinguishable from a strong one. In this case a person cannot know how much confidence they should actually have in a specific memory.

This is consistent with the general observation that memory accuracy has little connection to a person's confidence in that memory.[14]

[14] Simons et al. (2010); Chua et al. (2004); and Leippe (1980)

3.2 Benefits of Re-encoding

Because information is destroyed by the re-encoding process, re-encoding has a significant cost, but it can also have substantial benefits. This is because many memory systems move from initial "stable" memory states to much less stable ones over time. Here "stability" refers to a tendency for the memory state to maintain a distinctly recognizable origin rather than the tendency for the state to remain exactly the same.

Consider the extreme case of the absorbing star with arm length k. Over time, the memory state moves toward the absorbing middle of the star, but it can only move one place per period. The original state only becomes unrecognizable if the memory state reaches the absorbing state. If one were to re-encode every $k-1$ periods, the memory would never reach the absorbing state, and an agent with this memory system would never make a mistake.

One could conceivably construct environments where memory states tend to become more stable over time. For example, say that memories are encoded as $\tilde{m}_\theta$. Each period, the memory state transitions to the uninformative m_0 with probability 0.25. With probability 0.75 it transitions to the correct vault state m^*_θ. Memories in m_0 and m^*_θ do not transition. In this setting re-encoding only causes problems. However such environments are rarely realistic representation of actual memory systems, as they generally imply a very poorly chosen initial encoding. In the vaulting example, it is obviously better to initially encode memories with the vault states.

This brings the question of precisely when re-encoding can be helpful

Proposition 2. *Given, simplex like memory, uniform prior, and matching-based utility, for at least one Bayesian re-encoding to be weakly performance improving for some τ, it is necessary and sufficient to show that $\exists \tau_1, \tau_2$ such that $\tau_1 + \tau_2 = \tau$ and $\rho(\tau_1+\tau_2) \leq \frac{|\Theta|}{|\Theta|-1}\rho(\tau_1)\rho(\tau_2) + \frac{1}{|\Theta|-1}\left(1-\rho(\tau_1)-\rho(\tau_2)\right).$*

For proof see Appendix A.2.

This is a somewhat uncommon condition to deal with, so we also provide a more standard sufficient condition for re-encoding to be beneficial

Corollary 1. *Given, simplex like memory, uniform prior, and matching-based utility, if $\rho(\bullet)$ is (strictly) concave in some positive neighborhood of $t = 0$, there exists a (strictly) performance improving Bayesian re-encoding scheme.*

For proof see Appendix A.3.

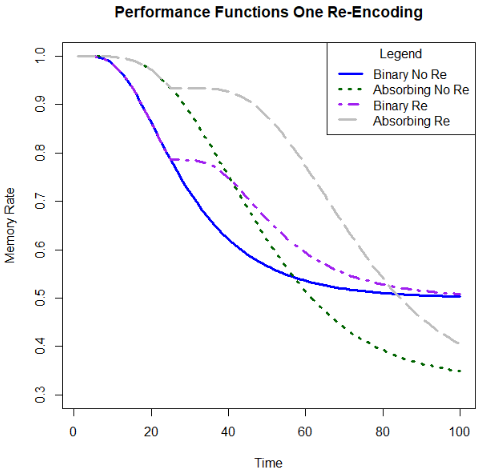

Figure 4: Performance functions with and without re-encoding. Binary: $\delta = 0.03$, $n = 13$, $|\Theta| = 2$. Absorbing Star: $\delta = 0.1, k = 3, |\Theta| = 3$. Re-encoding at $t = 25$.

Based on this corollary, it is useful to consider what types of memory systems have concave $\rho(t)$ near $t = 0$. Because $\rho(t) \in [0, 1]$ and is decreasing, it is generally not everywhere concave. However, many interesting $\rho(t)$s do have significant concave regions when t is small. Both the two state binary encoding and the absorbing star memory systems have a decreasing logistic shape with a concave early portion. In these cases, as long as it occurs in the concave regions, a re-encoding will be beneficial. Figure 4 shows the performance functions for both of the example memory systems with and without re-encoding.

3.2.1 Recall Effects

We have established that re-encoding a memory can improve memory performance. If we assume that explicitly recalling a memory will cause the brain to re-encode that memory, then this benefit is consistent with the observed benefits of forced recall where explicitly making someone recall a memory improves later recall.[15] The may also explain why the brain seems to “replay” events as part of the memory formation process.[16]

Spacing effects are phenomena where learning in a spaced out schedule provides better results than more clustered studying.[17] There are a number of spacing effects, but we are specifically concerned with the spaced rehearsal effect where an instance of learning involves rehearsing information by one’s

[15] Brewer et al. (2010)
[16] Buhry et al. (2011)
[17] Leicht and Overton (1987)

self with no new information provided.[18] Spacing out these rehearsals leads to better results than clustering them.

The way that re-encoding improves memory performance can explain why we observe this effect. If multiple Bayesian re-encodings happen in rapid succession, few errors remain for the second one to correct, essentially wasting the effort of the re-encoding. Regularly spacing re-encodings works better than clustering them.

When bayesian re-encodings are evenly spaced, we can use the resulting transition kernel to find a closed form solution to the final performance.

Proposition 3. *Given, simplex like memory, uniform prior, and matching-based utility, when there are r evenly spaced re-encodings occur, final performance at time T is given by*

$$\tfrac{1}{|\Theta|}\left(1+(|\Theta|-1)\left(\rho(\hat{\tau})-\tfrac{1-\rho(\hat{\tau})}{|\Theta|-1}\right)^{r+1}\right)$$

Where $\hat{\tau}=\frac{\tau}{r+1}$.

For proof see Appendix A.4. Note, as $|\Theta| \to \infty$ this performance approaches $\rho(\hat{\tau})^{r+1}$, because cases where two mistakes lead to a correct answer become effectively impossible.

This result will be very helpful in exploring the realism of memory decay rate in the next section.

3.3 Empirical Memory Performance

Having shown that Bayesian re-encoding can explain rehearsal effects, we turn our attention to the empirically observed rate of memory decay. In the experiments of Averell and Heathcote (2011), subjects memorized words and then attempted to recall them at several later times with no feedback between recall attempts. This allowed Averell and Heathcote (2011) to trace a retention curve which they checked against several candidate functions in order to determine its approximately determine its shape.

We argue that their retention curve is analogous to the performance function in our model. While Averell and Heathcote (2011) do not reward subjects based on performance, the matching reward seems like a reasonable approximation of intrinsic reward. Therefore we will compare their results to the predictions from our model.

[18] A spacing effect has also been observed in studying environments where information is repeated regularly. The re-encoding model does not have anything to say about those settings.

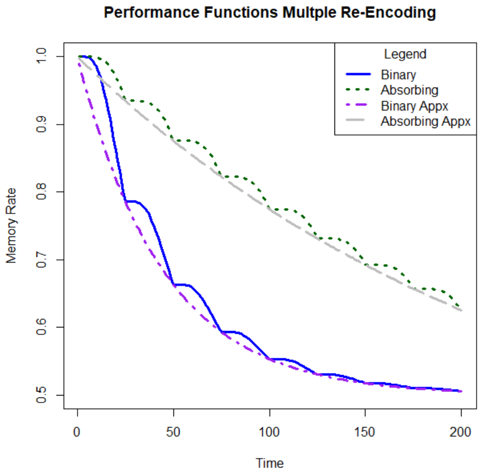

Figure 5: Performance given multiple re-encodings with approximation curves. Binary: $\delta = 0.03$, $n = 13$, $|\Theta| = 2$. Absorbing Star: $\delta = 0.1, k = 3, |\Theta| = 3$. Re-encoding every 25 periods.

Performance curves in Averell and Heathcote (2011) appear to follow an everywhere convex, approximately power law decay. This differs substantially from what we would expect in a well designed memory system. As discussed in Section 3.2, if the encoding is well chosen, memories should be initially encoded in relatively "stable" memory states. The transition to less stable states over time should lead to a concave region in the performance curve and an increase in the hazard rate. However, data from Averell and Heathcote (2011) shows that the hazard rate of human memories is either constant or decreasing over the entire range.

Re-encoding can explain this discrepancy. Unlike the transitions between memory states through decay, the transitions between encoding states through re-encoding have a constant hazard rate. To see this we modify the expression from Proposition 3 from having a fixed end point to instead give the approximate performance function when re-encoding occurs every k periods

$$\rho(t) \approx \tfrac{1}{|\Theta|}\left(1 + (|\Theta| - 1)\left(\rho(k) - \tfrac{1-\rho(k)}{|\Theta|-1}\right)^{t/k}\right)$$

which is a power law decay offset by a constant $\frac{1}{|\Theta|}$. Figure 5 shows graphically how frequent re-encoding will cause the performance function to approximate power law decay.

We have now shown that Bayesian re-encoding explains a range of human memory features previously thought to be separate. We now move on to a more general theoretical discussion of the properties of re-encodings.

4 General Memory Structure

In this section we present a number of results which will help us better understand Bayesian re-encoding outside of simplex-like memory systems. Ultimately, we will by trying to generalize the performance function in a broader context. We assume some arbitrary memory system defined by D and N and arbitrary prior over states π. We still assume a matching based utility function.

Before we get to the primary results characterize a number of useful objects. The memory system induces a sequence of signal structures on the state θ with earlier signal structures Blackwell dominating those that come later. We represent the signal structures as a distribution S_t of signals conditional on states. With no re-encodings, we have $S_t = D^t N$. We can also represent it as a stochastic matrix where $S_t[i, j]$ is the probability of receiving signal indexed i in state j.

Lemma 1. *For any sequence of posteriors defined by* $\{S_t : t = 1, 2, ...\tau\}$ *where* $S_t \succeq_B S_{t+1}$ *there exists a memory system* $\mathcal{M}, D$ *which induces it.*

For proof see Appendix A.5. Note this Lemma does not depend on matching based utility. The proof is fairly straightforward given the observation that the memory system can be set up to visit a different set of memory states in each time period. Given this lemma, the memory system at time t could, taken in isolation, induce generate any valid signal structure.

This Lemma shows us that our model does not just apply to memory systems, since the memory component of the model has little bite on its own. Re-encoding can be applied to any environment where information is passed through a series of noisy channels as long as their are opportunities to make the re-encodings. As such, our results could also have application in fields like telecommunications and technological diffusion.

4.1 The Confusion Matrix

Next we define an important object which is derived from the induced signal structure: the confusion matrix. The confusion matrix tells us how likely it is to end up in the encoding state for state θ_i after re-encoding given that we started in the encoding state for θ_j. As an object, it is critically important for calculating the performance function after re-encodings, because it essentially provides the transition matrix between re-encoding states induced by the re-encoding process.

The confusion matrix $Q(t, \mathcal{M}, D)$ is a $|\Theta| \times |\Theta|$ column stochastic matrix where the ijth element denotes the probability that $\bar{\gamma}(m) = \theta_i$ given $\theta = \theta_j$ where $\bar{\gamma}(m)$ is the mode of posterior the $\gamma(m)$.

We suppress the dependence on $\mathcal{M}$ and D since they are generally fixed. Formally

$$Q(t)[i,j] = \sum_{k:\theta_i=\bar{\gamma}(m_k)} S[k,j]$$

The concept of a confusion matrix makes sense in context of Bayesian re-encoding specifically, because the re-encoding depends on the posterior mode. The no re-encoding performance function is given by

$$E(u) = Tr(Q(T)\Pi)$$

So we know $Q(T) = AD^T N$ with A chosen optimally based on posterior mode. We can now apply several existing results to characterize $Q(\bullet)$.

Remark 2. Any $|\Theta| \times |\Theta|$ column stochastic matrix satisfying M satisfying $M_{ii}\pi(\theta_i) \geq M_{ij}\pi(\theta_j)\forall i,j$ can be a confusion matrix $Q(\tau)$ for some memory system given arbitrary τ.

This remark is an immediate result of the No Improving Action Switches result of Caplin and Martin (2015) (which holds iff) and Lemma 1. It says that any appropriately sized stochastic matrix with a type of prior dependent diagonal dominance is a possible confusion matrix.

While the model does not impose further restrictions on $Q(t)$ in isolation, it does impose dynamic restrictions. For example, by Blackwell (1953) and Lemma 1 we know $Tr(Q\Pi)$ is decreasing in t. Is it possible to impose greater restrictions on $Q(t)$ given the Blackwell ordering of the signal structures induced? At least in the 2×2 case, the answer is no.

Remark 3. In the two state case any sequence of valid $Q(t)$s with weakly decreasing $Tr(Q\Pi)$ is possible

For proof see Appendix A.6. Here valid means satisfying the NIAS condition. This remark comes from the fact that there exists a signal structure which can produce any valid $Q(t)$ with a given $Tr(Q\Pi)$ and this matrix can be garbled to given lower utility versions of itself.

Unfortunately, no such generating signal structure exists in the three state case, and further dynamic restrictions on $Q(t)$ in larger environments have so far been difficult to fully characterize. Note, in the previously discussed Simplex-Like memory systems the confusion matrix always takes a convenient form with a value q on the diagonal and the value $\frac{1-q}{|\Theta|-1}$ everywhere else. As τ increases, q weakly decreases. However, even in this simple case we generally have $Q(\tau) \neq Q(1)^\tau$. As we showed before, the rate of decay is generally not consistent.

4.2 Generalized Performance Function

Having defined and characterized the confusion matrix, we now have what we need to start discussing the performance function in the more general settings. Using the same logic as Proposition 3 we get the following result

Proposition 4. *Given matching-based utility, when there are r evenly spaced re-encodings, and*

$$\arg\max_j \boldsymbol{e}_i^T D^{\frac{\tau}{r+1}} N \Pi \boldsymbol{e}_j =$$

$$\arg\max_j \boldsymbol{e}_i^T D^{\frac{\tau}{r+1}} N Q\left(\tfrac{\tau}{r+1}\right)^k \Pi \boldsymbol{e}_j$$

for all for all $i, k \leq r+1$ then final performance at time τ is given by

$$Tr\left(Q\left(\tfrac{\tau}{r+1}\right)^{r+1} \Pi\right)$$

Proof. The result is immediate as long as the confusion matrix is constant based on the logic behind Proposition 3. For the confusion matrix to be constant, it must be that the same memory states at the same time after must give induce the same posterior mode at each re-encoding step and at decision time. The condition guarantees this consistency through Bayes rule.

This Proposition gives us a simple formula for the performance function under Bayesian re-encoding as long as the posterior mode after seeing a given memory state remains the same regardless of how many Bayesian re-encodings have occurred. Note that this condition hold trivially if the player cannot remember how many Bayesian re-encodings have happened.[19] Mathematically this restriction boils down to preserving a type of diagonal dominance across multiplications by the confusion matrix.

The condition holds for any simplex-like memory system (as mentioned above) and tends to have more slack when $Q(t)$ is highly symmetric and memory signals are more informative. This symmetry does not need to extend as far as that seen in Simplex-Like memory systems. We can take advantage of a less strenuous form of symmetry. First we provide a symmetry definition for the decay matrix

Definition 4. A $K \times K$ Matrix is a ring-distance monotone if it has the form $\begin{bmatrix} v & v_{r.1} & v_{r.2} & \cdots \end{bmatrix}$ where v is a vector of length K which satisfies $v[i] = v[K+2-i] \forall i \geq 2$ and $v[i]$ is decreasing in i for

[19] This does make calculating the confusion matrix more difficult, as the player will have to hold beliefs over the number of re-encoding that have happened.

all $i \leq \frac{K}{2}+1$. Here $v_{r.i}$ denotes v rotated i elements so $v[1] = v_{r.1}[2]$, $v[2] = v_{r.1}[3]$, and so on until $v[K] = v_{r.1}[1]$.

We call this ring-distance monotone, because treating the matrix as a weighted circular graph, it implies that connections monotonically and consistently decrease in strength the farther two nodes are away from each other on the circle.

Next we need a symmetry condition for the encoding matrix

Definition 5. An encoding $k|\Theta| \times |\Theta|$ matrix N is evenly spaced if $N[i,j] = 1$ if $(i-1)*k+1 = j$ and 0 otherwise.

This evenly spaced encoding matrix is essentially an identity matrix that has been stretched out with lots of 0s. It is called evenly spaced because it spaces the encoding memory states evenly around the circle. Now we can write the result.

Corollary 2. *Given matching utility and uniform prior, if D is ring-distance monotone, N is evenly spaced, and π is uniform then if here are r evenly spaced re-encodings performance at time τ is given by*

$$\tfrac{1}{|\Theta|} Tr\left(Q\left(\tfrac{\tau}{r+1}\right)^{r+1}\right)$$

Where $Q\left(\frac{\tau}{r+1}\right) = AD^{\frac{\tau}{r+1}}N$ and $A[i,j] = 1$ if $ik - 1.5k + 1 < j < ik - 0.5k + 1$, $A[i,j] = 0.5$ if $j = ik - 0.5k + 1$ or $j = ik - 1.5k + 1$, and 0 otherwise.

For Proof see Appendix A.7. The essence of this proof boils down to showing that ring-distance monotone matrices are diagonal dominant and closed under multiplication. Note that this also works for re-indexings of ring-distance monotone matrices, since which node in a network is labeled as i inherently arbitrary.

This concludes the results related to Bayesian re-encoding. Next we will be considering re-encoding more generally.

5 Optimal Re-encodings Beyond Bayes

So far we have been considering Bayesian Re-encoding, without considering whether that types of re-encoding is optimal. We have also been only considering environments where there is only a payoff for matching one's action with the state of the world, because Bayesian re-encoding makes sense in these contexts.

We now consider re-encodings more generally and what we can say about optimal re-encodings. Ultimately, we will show why Bayesian re-encodings are optimal in the examples we previously considered. We also generalize from the memory environment by replacing powers of the memory decay matrix D^τ with arbitrary sequences of garblings $\hat{D}_i$. What we previously called memory states, we now call signal states. We still consider a setting with finite states, actions, and signal realizations. We now allow for arbitrary utility functions.

We need to adapt 1 to deal with general utility functions. The probability of action conditional on state is given by $A\bar{D}$ where $\bar{D}$ is the total channel garbling after passing through all garbling in a sequence. The joint action, state probabilities are given by

$$A\bar{D}N\Pi$$

This is essentially the same approach as Leshno and Spector (1992) but transposed to show that by pre-multiplying stochastic matrices we are applying repeated transformations. Given this, the player's expected utility is

$$\rho(\tau) = \boldsymbol{e}^T \left(\left(A\bar{D}N\Pi \right) \odot U \right) \boldsymbol{e}$$

Where $\odot$ is the Hadamard product and $U[i,j] = u(a_i, \theta_j)$. With these preliminaries out of way we can turn our attention to re-encodings.

5.1 Single Re-encoding

To ad a re-encoding we first separate the decay into two phases as

$$\boldsymbol{e}^T \left(\left(A\hat{D}_2\hat{D}_1 N\Pi \right) \odot U \right) \boldsymbol{e}$$

Where $\bar{D} = \hat{D}_2\hat{D}_1$. If we include re-encoding between the phases this becomes

$$\boldsymbol{e}^T \left(\left(A\hat{D}_2 R\hat{D}_1 N\Pi \right) \odot U \right) \boldsymbol{e}$$

Where $R[i,j]$ is the probability of re-encoding the signal state indexed i after seeing the memory state indexed j. Define $\mathcal{R}$ as the space of $|\mathcal{M}| \times |\mathcal{M}|$ stochastic matrices.

In this context, the benefit re-encoding seems counterintuitive. Re-encoding is, essentially, another garbling. We know

$$G_1 \succeq_B G_2 G_1$$

where G_1 and G_2 are arbitrary stochastic matrices, so it is intuitive that adding more garblings and mappings should make things worse. However, this only applies to post-multiplying or pre-multiplying. It does not apply when inserting a new mapping in the middle. In general,

$$G_2 G_1 \not\succeq_B G_2 R G_1$$

Where R is a re-encoding mapping. In some cases $G_2 R G_1$ can Blackwell dominate $G_2 G_1$. This shows how re-encoding can be beneficial.

The optimal re-encoding problem can then be written as

$$R^* = \arg\max_{R \in \mathcal{R}} \boldsymbol{e}^T \left(\left(A \hat{D}_2 R \hat{D}_1 N \Pi \right) \odot U \right) \boldsymbol{e}$$

Then optimal R^*can be found using the following proposition

Proposition 5. *If A and N are fixed, it is necessary and sufficient for R to be optimal that if $R[i,j]$ is positive then*

$$j = \arg\max_k M[k,i]$$

Where $M = \hat{D}_1 N \Pi U^T A \hat{D}_2$

For Proof see Appendix A.8. If the matrix M can be calculated, finding the optimal re-encoding is as simple finding the column maximum. Note that Similar logic can be used to find optimal N with A and R fixed or optimal A with N and A fixed.

This result is strong but somewhat narrow. Next we will see how the same logic can be extended to environments where A and N aren't fixed or where there are multiple re-encodings

5.2 General Re-encoding Plan

Note that while the sufficiency no longer applies, a similar condition is necessary. In this case the optimal re-encoding problem becomes

$$\arg \max_{A,N,R_1,R_2,...} \boldsymbol{e}^T \left(\left(A\hat{D}_{\tau-1}R_{\tau-2}...R_2\hat{D}_2R_1\hat{D}_1N\Pi \right) \odot U \right) \boldsymbol{e}$$

Where the $\hat{D}_i$s are all some sequence of garblings. We now extend Proposition 5, although we lose sufficiency in doing so.

Corollary 3. *It is necessary for the re-encoding plan to be optimal that if $R_t[i,j]$ is positive then*

$$j = \arg\max_k M_t[k,i]$$

For all t. Where $M_t = \hat{D}_t\hat{D}_{t-1}...\hat{D}_2R_1\hat{D}_1N\Pi U^T A\hat{D}_{\tau-1}R_{\tau-2}...R_{t+1}\hat{D}_{t+1}$.

Proof is omitted as it is essentially identical to the sufficiency in the proof of Proposition A.8 with extra garblings and re-encodings. Note A and N essentially behave as, R_{T-1} and R_0 respectively.

This result can be helpful in finding local optima but it does not give us a way to search for the globally optimal set of re-encodings easily.

However, there are a few results which can make searching for optimal re-encodings easier in many contexts. First we have the following remark:

Remark 4. A deterministic optimal re-encoding exists

For proof see Appendix A.9. This means that the search space will be finite, if still potentially large.

We can narrow the search space of re-encoding plans further by discarding some immediately suboptimal re-encodings. First we need another definition.

Definition 6. We say a consecutive sequence of re-encodings $\{R_t, R_{t+1}, R_{t+2}, ..., R_{t+n}\}$ is effectively Blackwell dominated if $\exists\{R^{'}_t, R^{'}_{t+1}, R^{'}_{t+2}, ..., R^{'}_{t+n}\}$ such that $M = GM'$ or $M\hat{D}_t = GM'\hat{D}_t$ for some stochastic matrix G, where $M = \hat{D}_{t+n+1}R_{t+n}...\hat{D}_{t+2}R_{t+1}\hat{D}_{t+1}R_t$ and $M' = \hat{D}_{t+n+1}R^{'}_{t+n}...\hat{D}_{t+2}R^{'}_{t+1}\hat{D}_{t+1}R^{'}_t$.

Now we can categorize some re-encodings that will never be used:

Proposition 6. *An effectively Blackwell dominated re-encoding sequence (potentially including initial encoding) will always be weakly suboptimal*

For proof see Appendix A.10. This is an intuitive result, because it never makes sense to strictly give up information one could have kept through a different choice of re-encodings. For example, it never makes sense to re-encode all current signal states onto the same signal state, as doing so destroys all the available information.

Note that this Proposition can be leverage show the optimality of the employed encodings and Bayesian re-encoding in the twostate binary and absorbing star memory systems discussed previously.

6 Potential for Applications

In this paper we proposed a new mathematical framework for re-encoding as a valuable technique when dealing with information repeatedly passing through noisy channels. This is particularly applicable when dealing with memory systems. We showed that re-encoding can improve memory performance and explain a number of quirks in human memory.

We now consider a few current and potential future applications of re-encoding. To our knowledge, nothing like Bayesian re-encoding is used in telecommunication, and the need for encryption would make it somewhat impractical. For Bayesian re-encoding to work, intermediaries need access to the information in the message. Most error correction is instead done by simply re-sending lost or damaged packets. Bayesian re-encoding could be valuable in situations where encryption is not required.

Usage in memory systems is generally rarer. As mentioned, a limited form of re-encoding called Error Correction Code (ECC) memory fixes small, usually single bit errors, but it is not in common use, and it is not on the scale of whole files or whole memories.[20] There are a number of reasons for this.

First, computer memory mediums are extremely robust with most devices not experiencing a single DRAM bit flip error in a given year unless they are exposed to unusual amounts of radiation.[21] This means that individual bits are robust relative to the durability of the medium they exist on. A hard-drive is likely to fail from use before a specific bit gets an error. With brains the line between the data content and the data medium is blurrier, but the brain's operational lifetime is much greater than the persistence time of an individual synaptic connection.

Also, processing power is at a relatively greater premium than storage space on modern computers, so making backups makes more sense than regular re-encoding. However, both of these trade-offs are

[20] Chen and Hsiao (1984)

[21] Schroeder et al. (2009)

specific to the technologies of today. If later computers more closely resemble neural and biological systems, re-encoding may play a critical roll in those systems as well. Already, devices like Intel's Loihi Neuromorphic Processor are moving in this direction.

Biological systems use a complex and broad array of processes to manage errors in DNA, but two of the mechanisms involved can be thought of as roughly analogous to ECC and active Bayesian re-encoding respectively. DNA information is stored redundantly as in ECC memory with each strand containing identical information. When on strand is damaged, it can often be repaired based on the other strand through nucleotide excision repair.[22] This is similar to how ECC uses redundant coding to repair errors. However, in the case of more serious double strand errors, the cell may use homologous recombination to replace the damaged gene with the equivalent gene from the matching chromosome.[23] This process could be considered more similar to Bayesian re-encoding, since the cell is essentially replacing the original gene with a "best guess" of the original contents.

To our knowledge re-encoding has not yet been applied in any social science context.

References

Averell, L. and Heathcote, A. (2011). The form of the forgetting curve and the fate of memories. *Journal of Mathematical Psychology*, 55(1):25–35.

Blackwell, D. (1953). Equivalent comparisons of experiments. *The Annals of Mathematical Statistics*, 24(2):265–272.

Brewer, G., Marsh, R., Clark-Foos, A., and Meeks, J. (2010). Noncriterial recollection influences metacognitive monitoring and control processes. *The Quarterly Journal of Experimental Psychology*, 63(10):1936–1942.

Buhry, L., Azizi, A., and Cheng, S. (2011). Reactivation, replay, and preplay: how it might all fit together. *Neural Plasticity*.

Caplin, A. and Martin, D. (2015). A testable theory of imperfect perception. *The Economic Journal*, 125(582):184–202.

Chen, C. L. and Hsiao, M. Y. (1984). Error-correcting codes for semiconductor memory applications: A state-of-the-art review. *IBM Journal of Research and Development*, 28(2):124–134.

[22] Fuss and Cooper (2006)

[23] Li and Heyer (2008)

Chua, E., Rand-Giovanetti, E., and Sperling, R. (2004). Dissociating confidence and accuracy: Functional magnetic resonance imaging shows origins of the subjective memory experience. *Journal of Cognitive Neuroscience*.

Ebbinghaus, H. (1885). Memory: A contribution to experimental psychology. *Classics in the History of Psychology*. An internet resource developed by Christopher D. Green. York University, Toronto, Ontario.

Fuss, J. O. and Cooper, P. K. (2006). Dna repair: dynamic defenders against cancer and aging. *PLoS ONE Biology*, 4(6).

Hemmer, P. and Steyvers, M. (2009). A bayesian account of reconstructive memory. *Topics in Cognitive Science*, 1(1):189–202.

Jacoby, L. L. (1978). On interpreting the effects of repetition: Solving a problem versus remembering a solution. *Journal of Verbal Learning and Verbal Behavior*, 17(6):649–667.

Leicht, K. and Overton, R. (1987). Encoding variability and spacing repetitions. *The American Journal of Psychology*, 100(1):61–68.

Leippe, M. (1980). Effects of integrative memorial and cognitive processes on the correspondence of eyewitness accuracy and confidence. *Law and Human Behavior*, 4(4):261–274.

Leshno, M. and Spector, Y. (1992). An elementary proof of blackwell's theorem. *Mathematical Social Sciences*, 25(1):95–98.

Li, X. and Heyer, W.-D. (2008). Homologous recombination in dna repair and dna damage tolerance. *Cell Research*, 18:99–113.

Lin, H.-Y., Moser, S., and Chen, P.-N. (2018). Weak flip codes and their optimality on the binary erasure channel. *IEEE Transactions on Information Theory*, 64(7):5191–5218.

Mulligan, N. and Peterson, D. (2014). The negative testing and negative generation effects are eliminated by delay. *Journal of experimental psychology. Learning, memory, and cognition*, 41.

Murre, J. M. J. and Dros, J. (2015). Replication and analysis of ebbinghaus forgetting curve. *PLOS ONE*, 10(7).

Roediger, H. and McDermott, K. (1995). Creating false memories: Remembering words not present in lists. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 21(4):803–814.

Schacter, D. (2001). *The seven sins of memory: How the mind forgets and remembers.* Houghton, Mifflin and Company, Boston, MA, US: Houghton, Mifflin and Company.

Schroeder, B., Pinheiro, E., and Pinheiro, E. (2009). Dram errors in the wild: A large-scale field study. In *SIGMETRICS*.

Simons, J., Peers, P., Mazuz, Y., Berryhill, M., and Olsen, I. (2010). Dissociation between memory accuracy and memory confidence following bilateral parietal lesions. *Cerebral Cortex*, 20(2):479–485.

Stark, C. E., Okado, Y., and Loftus, E. F. (2010). Imaging the reconstruction of true and false memories using sensory reactivation and the misinformation paradigms. *Learning and Memor7*, 17:485–488.

A Proofs

A.1 Proof of Proposition 1

The fact that the conditional memory state probabilities depend only on the previous encoding is immediate from the decay process, so we must only need to show that the mapping from memory state to actions does not depend on the number of re-encoding steps when the memory system is simplex-like.

Since information will always weakly improve performance of the agent, $\rho(\theta, T)$ must be greater than the probability of choosing $a = \theta$ when guessing with no memory signal. Recall, in the no re-encoding case

$$a(m,t) = \arg\max_{\theta} \gamma(\theta|m,t)$$

Note that due to the uniform prior, this becomes $\theta_{i(m_j,t)}$ where

$$i(m_j,t) = \arg\max_{i} D^t N \boldsymbol{e}_i[j]$$

After some number of re-encodings, n, the action function $a_R(m,t,n)$ becomes $\theta_{i(m_j,t,n)}$ where

$$i(m_j, t, n) = \arg\max_i \sum_k p\left(\hat{\theta}_n = \theta_i | \hat{\theta}_0 = \theta_i\right) D^t N \boldsymbol{e}_i[j]$$

Where $\hat{\theta}_n$ is the state encoded be re-encoding n and $\hat{\theta}_0$ is the initial encoding state. Note that $D^t N\boldsymbol{e}_i[j]$ takes on the same values across the sum, regardless of $\hat{\theta}_0$. By the simplex-like property of the memory system, we know that $p\left(\hat{\theta}_n = \theta_i | \hat{\theta}_0 = \theta_i\right)$ only takes two values which we will call a, b.

$$p\left(\hat{\theta}_n = \theta | \hat{\theta}_0 = \theta\right) = a$$

$$p\left(\hat{\theta}_n = \theta' | \hat{\theta}_0 = \theta\right) = b \forall \theta' \neq \theta$$

Note that $a \geq b$ by the fact that an informed agent will always perform weakly better than chance. Therefore, the optimal $i(m_j, t, n)$ is therefore the one that matches the high value of $p\left(\hat{\theta}_n = \theta_i | \hat{\theta}_0 = \theta_i\right)$ to the highest value in $D^t N\boldsymbol{e}_i[j]$ which is $i(m_j, t, n) = i(m_j, n)$. Therefore, $a_R(m, t, n) = a(m, t)$ □

A.2 Proof of Proposition 2

Note that, as shown in the Proof of Proposition 1, the mapping from memory state to actions does not depend on the number of re-encoding steps when the memory system is simplex-like.

We cover the "if" and "only if" parts separately.

If: Consider a decision setting $\tau = \tau_1 + \tau_2$. Performance with no re-encodings would be $\rho(\tau_1 + \tau_2)$. If we introduce one re-encoding at τ_1, performance becomes $\rho(\tau_1)\rho(\tau_2) + (1 - \rho(\tau_1))(1 - \rho(\tau_2))\frac{1}{|\Theta|-1} = \frac{|\Theta|}{|\Theta|-1}\rho(\tau_1)\rho(\tau_2) + \frac{1}{|\Theta|-1}\left(1 - \rho(\tau_1) - \rho(\tau_2)\right)$.

Only if: say there is an optimal re-encoding scheme that involves at least one re-encoding. The first re-encoding is at t_1 and the second re-encoding or τ is at t_2. From Proposition 1, we know that performance at τ is weakly increasing in performance at t_2. Define $\tau_1 = t_1$ and $\tau_2 = t_2 - t_1$. The performance at t_2 under the optimal re-encoding scheme is $\frac{|\Theta|}{|\Theta|-1}\rho(\tau_1)\rho(\tau_2) + \frac{1}{|\Theta|-1}\left(1 - \rho(\tau_1) - \rho(\tau_2)\right)$. If $\rho(\tau_1 + \tau_2) > \frac{|\Theta|}{|\Theta|-1}\rho(\tau_1)\rho(\tau_2) + \frac{1}{|\Theta|-1}\left(1 - \rho(\tau_1) - \rho(\tau_2)\right)$ it would improve performance at t_2 to remove the re-encoding at t_1. This contradicts optimality of the re-encoding scheme. □

A.3 Proof of Corollary 1

Note if ρ is concave then

$\rho(\tau_1+\tau_2)-\rho(\tau_0)\leq\rho(\tau_1)-\rho(\tau_0)+\rho(\tau_2)-\rho(\tau_0)$ for $\tau_0\leq\tau_1,\tau_2$

Say $\tau_0=0$ and recall $\rho(0)=1$, so $p(\tau_1+\tau_2)\leq\rho(\tau_1)+\rho(\tau_2)-1$

So it suffices to show

$\rho(\tau_1)+\rho(\tau_2)-1\leq\frac{|\Theta|}{|\Theta|-1}\rho(\tau_1)\rho(\tau_2)+\frac{1}{|\Theta|-1}\left(1-\rho(\tau_1)-\rho(\tau_2)\right)$

which can be rewritten

$\rho(\tau_2)(1-\rho(\tau_1))\leq1-\rho(\tau_1)$

We know $1-\rho(\tau_1)\geq0$ and $\rho(\tau_2)\geq0$, so this holds. If $\rho(\bullet)$ is strictly decreasing at 0 $1-\rho(\tau_1)>0$, so it holds strictly. A strictly concave decreasing function is strictly decreasing. □

A.4 Proof of Proposition 3

Note that for exact even spacing to be possible, $\hat{\tau}=\frac{\tau}{r+1}$ must be an integer. Define $q=\rho\left(\hat{\tau}\right)$. Therefore, the transition matrix between re-encodings is then

$$G=\begin{bmatrix} q & \frac{1-q}{|\Theta|-1} & \frac{1-q}{|\Theta|-1} & \dots \\ \frac{1-q}{|\Theta|-1} & q & \frac{1-q}{|\Theta|-1} & \dots \\ \frac{1-q}{|\Theta|-1} & \frac{1-q}{|\Theta|-1} & q & \dots \\ \dots & \dots & \dots & \dots \end{bmatrix}$$

So the probability of a correct response at time τ is $\frac{1}{|\Theta|}tr(G^{\hat{\tau}})$ as long as $\hat{\tau}$ is an integer. We can find $G^{\hat{\tau}}$ by diagonalizing G and iterating that way. To do that we need the following lemma.

Lemma 2. *G has two eigenvalues, $q-\frac{1-q}{|\Theta|-1}$ which is repeated $|\Theta|-1$ times and 1 which is only repeated once.*

The corresponding eigenvectors form a basis

$$B=\begin{bmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ -1 & 0 & 0 & 0 & \dots & 1 \\ 0 & -1 & 0 & 0 & \dots & 1 \\ 0 & 0 & -1 & 0 & \dots & 1 \\ 0 & 0 & 0 & -1 & \dots & 1 \\ \dots & \dots & \dots & \dots & \dots & 1 \end{bmatrix}$$

Proof of Lemma 1. We can verify the first $|\Theta|-1$ eigenvalue, eigenvector pairs by checking

$$\left(G-\left(q-\frac{1-q}{|\Theta|-1}\right)I\right)\boldsymbol{v}_i=\mathbf{0}\forall i\in\{2,3,...,|\Theta|\}$$

where $\boldsymbol{v}_i$is a vector with one in the first position, negative one in position i and zero in all other positions

$$\left(G-\left(q-\frac{1-q}{|\Theta|-1}\right)I\right)=\begin{bmatrix}\frac{1-q}{|\Theta|-1} & \frac{1-q}{|\Theta|-1} & \frac{1-q}{|\Theta|-1} & ...\\ \frac{1-q}{|\Theta|-1} & \frac{1-q}{|\Theta|-1} & \frac{1-q}{|\Theta|-1} & ...\\ \frac{1-q}{|\Theta|-1} & \frac{1-q}{|\Theta|-1} & \frac{1-q}{|\Theta|-1} & ...\\ ... & ... & ... & ...\end{bmatrix}$$

We can see immediately that multiplying this by and of the $\boldsymbol{v}_i$s will yield $\mathbf{0}$. Next we must check the pair 1, $\tilde{\boldsymbol{v}}$where $\tilde{\boldsymbol{v}}$is a length $|\Theta|$ vector of all ones. This means we must check

$$(G-I)\,\tilde{\boldsymbol{v}}=\mathbf{0}$$

First note

$$(G-I)=\begin{bmatrix}q-1 & \frac{1-q}{|\Theta|-1} & \frac{1-q}{|\Theta|-1} & ...\\ \frac{1-q}{|\Theta|-1} & q-1 & \frac{1-q}{|\Theta|-1} & ...\\ \frac{1-q}{|\Theta|-1} & \frac{1-q}{|\Theta|-1} & q-1 & ...\\ ... & ... & ... & ...\end{bmatrix}$$

Therefore $(G-I)\,\tilde{\boldsymbol{v}}$ is a vector with every element equal to $q-1+(|\Theta|-1)\left(\frac{1-q}{|\Theta|-1}\right)=0$. This concludes the proof of Lemma 1.

We now continue discussing the diagonalization of G. B can be inverted as

$$B^{-1}=\begin{bmatrix}\frac{1}{|\Theta|} & \frac{1}{|\Theta|}-1 & \frac{1}{|\Theta|} & \frac{1}{|\Theta|} & ...\\ \frac{1}{|\Theta|} & \frac{1}{|\Theta|} & \frac{1}{|\Theta|}-1 & \frac{1}{|\Theta|} & ...\\ \frac{1}{|\Theta|} & \frac{1}{|\Theta|} & \frac{1}{|\Theta|} & \frac{1}{|\Theta|}-1 & ...\\ \frac{1}{|\Theta|} & \frac{1}{|\Theta|} & \frac{1}{|\Theta|} & \frac{1}{|\Theta|} & ...\\ ... & ... & ... & ... & ...\end{bmatrix}$$

So $G^{\mu t}=B\Lambda^{\mu t}B^{-1}$ where $\Lambda=diag(q-\frac{1-q}{n-1},q-\frac{1-q}{n-1},q-\frac{1-q}{n-1},...,1)$

Multiplying this out gives a matrix $G^{\mu t}$ with diagonal elements

$$\tfrac{1}{|\Theta|}\left(1+(|\Theta|-1)\left(q-\tfrac{1-q}{|\Theta|-1}\right)^{\mu t}\right)$$

and off diagonal elements

$$\tfrac{1}{|\Theta|}\left(1-\left(q-\tfrac{1-q}{|\Theta|-1}\right)^{\mu t}\right)$$

So a player's accuracy is then $\tfrac{1}{|\Theta|}\left(1+(|\Theta|-1)\left(q-\tfrac{1-q}{|\Theta|-1}^{T/r}\right)\right)$. □

A.5 Proof of Lemma 1

The proof has two parts. First, we must show that S_1 is an arbitrary signal structure, then we must show S_{t+1} is an arbitrary garbling of S_t.

The first part is straightforward since D can provide an arbitrary mapping from the initial encoding states to other memory states.

The second part comes from the fact that one can construct memory systems such that they will visit disjoint sets of memory states each period (as with the absorbing star). One can then simply induce an arbitrary garbling between memory states visitable in period t and those visitable in period $t+1$. □

A.6 Proof of Remark 3

Take an environment with $\Theta = \{\theta_1, \theta_2\}$ and a prior of π for state θ_1. WLOG assume $\pi \geq 0.5$. Define the expected utility from the confusion matrix $Q_{11}\pi + Q_{22}(1-\pi) = u$. First we show that there exists a signal structure which can produce any valid confusion matrix. Then, we show that the signal structure can be garbled to the equivalent generator matrix for any lower u. Note we must have $u \geq \pi$ since memory provides an informative signal.

Part 1: Define an arbitrary legitimate confusion matrix with fixed u by

$$Q = \begin{bmatrix} Q_{11} & 1-\frac{u-Q_{11}\pi}{1-\pi} \\ 1-Q_{11} & \frac{u-Q_{11}\pi}{1-\pi} \end{bmatrix}$$

Consider the signal structure

$$S(u) = \begin{bmatrix} \frac{u-(1-\pi)}{\pi} & 0 \\ \frac{1-u}{\pi} & \frac{1-u}{1-\pi} \\ 0 & \frac{u-\pi}{1-\pi} \end{bmatrix}$$

This signal structure can produce any valid confusion matrix. Upon receiving signal 1, the player picks action θ_1. On receiving signal 3 they pick action θ_2. After receiving signal 2 the player picks θ_1 with probability λ and θ_2 with probability $1-\lambda$. We can produce any valid confusion matrix by setting

$$\lambda(Q_{11}) = \frac{Q_{11}\pi+(1-\pi)-u}{(1-u)}$$

Part 2: The second part of the result is immediate from the fact that

$$S(u') = \begin{bmatrix} \frac{u'-(1-\pi)}{u-(1-\pi)} & 0 & 0 \\ \frac{u-u'}{u-(1-\pi)} & 1 & \frac{u-u'}{u-\pi} \\ 0 & 0 & \frac{u'-\pi}{u-\pi} \end{bmatrix} S(u)$$

For all valid $u' < u$

A.7 Proof of Proposition 2

We begin with a lemma

Lemma 3. *If matrices X and Y are both $K \times K$ ring-distance monotone then XY is ring-distance monotone.*

Proof. Say $X = \begin{bmatrix} x \\ x_{r.1} \\ ... \\ x_{r.K-1} \end{bmatrix}$ and $Y = \begin{bmatrix} y & y_{r.1} & ... & y_{r.K-1} \end{bmatrix}$

$$\begin{bmatrix} x \\ x_{r.1} \\ ... \\ x_{r.K-1} \end{bmatrix} \begin{bmatrix} y & y_{r.1} & ... & y_{r.K-1} \end{bmatrix} = \begin{bmatrix} O_0 & O_{-1} & O_{-2} & ... & O_{-K+1} \\ O_1 & O_0 & O_1 & ... & O_{-K+2} \\ O_2 & O_1 & O_0 & ... & ... \\ ... & ... & ... & ... & O_{-1} \\ O_{K-1} & O_{K-2} & ... & O_1 & O_0 \end{bmatrix}$$

Where $O_i = x \cdot b_{r.i}$. Note under the wraparound structure ring-distance monotone imposes on X and Y $O_i = O_{K-i}$ and therefore $O_i = O_{-i}$ so this becomes

$$AB = \begin{bmatrix} O_0 & O_1 & O_2 & ... & O_1 \\ O_1 & O_0 & O_1 & ... & O_2 \\ O_2 & O_1 & O_0 & ... & ... \\ ... & ... & ... & ... & O_1 \\ O_1 & O_2 & ... & O_1 & O_0 \end{bmatrix}$$

Thus, if O_i is decreasing in i we have the result.

Define $\mathcal{C}_z(v) = (v[1], v[|v|], v[2], v[|v|-1], ...,)$ as the cyclic zipper of v. This reordering is created by turning the vector v into a circle and then "zippering" the two sides together.

We say a vector v of length K is a periodic ring-vector if it satisfies $v[i] = v[K+2-i] \forall i \geq 2$ and $v[i]$ is decreasing in i for all $i \leq \frac{K}{2}+1$. $v = (v_1, v_2, v_3, ...v_3, v_2)$ where $v_{i+1} \leq v_i$

Note that if v is a periodic ring-vector then $\mathcal{C}_z(v)$ is weakly decreasing. We also have the following property

Lemma 4. *If v is a periodic ring-vector of length K then $\mathcal{C}_z(rot_n(v))$ majorizes $\mathcal{C}_z(rot_{n+1}(v)) \forall n \leq K/2$*

To see this note that during such a rotation some elements of $\mathcal{C}_z(rot_n(v))$ move up in position along a specific path while others move down in along a different path. Define $c_n = \mathcal{C}_z(rot_n(v))$.

During rotation, from c_n to c_{n+1}, The elements $c_n[2], c_n[4], c_n[6], ...$ move up in position while the elements $c_n[1], c_n[3], c_n[5], ...$ move down. The second sequence elementwise dominates the first. This can be best seen from the accompanying diagram. Hence c_n majorizes c_{n+1}. □

We conclude the proof of Lemma 3 by noting that since $\mathcal{C}_z(\bullet)$ is a reordering then $a \cdot b = \mathcal{C}_z(a) \cdot \mathcal{C}_z(b)$. Since $\mathcal{C}_z(a)$ is weakly decreasing and $\mathcal{C}_z(b_{r.n})$ majorizes $\mathcal{C}_z(b_{r.n+1})$ then $a \cdot b_{r.n} \geq a \cdot b_{r.n+1}$. Thus O_i is decreasing in i concluding the proof. □

Next we need a result on the optimality of A with no re-encodings.

Lemma 5. *A is the optimal action matrix given a distance-ring decay matrix Y and an evenly spaced encoding matrix N.*

Proof. Note, for N to be evenly spaced it must be that Y is a $k|\Theta| \times k|\Theta|$ matrix where k is some integer. Given π uniform, this is the same as showing

$$\arg \min_{j\in\{1,\ldots|\Theta|\}} |i-(j-1)k-1| = \arg \max_{j\in\{1,\ldots|\Theta|\}} (YN)\,[i,j]\forall i$$

The greatest element of row i in YN corresponding to an encoding determines the posterior mode after memory state m_i. Using the evenly spaced encoding matrix, the encodings have coordinates $(j-1)k+1$ where j in an integer. This statement is saying that the most likely initial encoding in a row corresponds to the closest encoding coordinate to the rows index. Note this is precisely how A is constructed. Ties are split evenly.

To see this note $(YN)\,[i,j] = y_{r.i-1}\cdot \boldsymbol{e}_{(j-1)k+1}$. Given that y is a periodic ring vector, this gives the result. □

Note $D^{\frac{\tau}{r+1}}$is a distance-ring matrix by Lemma 3, so this will show that A is optimal for the first re-encoding. The last result we need shows another operation which preserves the distance ring nature of a matrix

Lemma 6. *If Y is a distance-ring matrix, then AYN is a distance ring matrix where N is an evenly spaced encoding matrix.*

Proof. Again note, for N to be evenly spaced it must be that Y is a $k|\Theta|\times k|\Theta|$ matrix where k is some integer.

Next see

$$(YN) = \begin{bmatrix} y & y_{r.1+k} & y_{r.+2k} & \cdots \end{bmatrix}$$

and

$$A = \begin{bmatrix} a \\ a_{r.1+k} \\ a_{r.1+2k} \\ \cdots \end{bmatrix}$$

Where a is a periodic ring vector. Therefore

$$AYN = \begin{bmatrix} O_0 & O_k & O_{2k} & ... & O_k \\ O_k & O_0 & O_k & ... & O_{2k} \\ O_{2k} & O_k & O_0 & ... & ... \\ ... & ... & ... & ... & O_k \\ O_k & O_{2k} & ... & O_k & O_0 \end{bmatrix}$$

Which has the distance-ring structure. □

Now by Lemma 5 we know that $Q\left(\frac{\tau}{r+1}\right) = AD^{\frac{\tau}{r+1}}N$ and by Lemma 6 we know Q is a distance ring matrix. By Lemma 3 this means D^m and Q^n are both distance ring matrices ∀ m and n. To make this work we implicitly assume D is an a $k|\Theta| \times k|\Theta|$ matrix where k is some integer.

So the last thing we need to prove is that

$$\arg \min_{j \in \{1,...|\Theta|\}} |i-(j-1)k-1| = \arg \max_{j \in \{1,...|\Theta|\}} \left(D^m N Q^n\right)[i,j] \forall i, m, n$$

Say

$$D^m = \begin{bmatrix} d^m & d^m_{r.1+k} & d^m_{r.+2k} & ... \end{bmatrix} = \begin{bmatrix} v^1 \\ v^2 \\ v^3 \\ ... \end{bmatrix}$$

Where v^i is every kth element of $d^m_{r.i-1}$ starting at 1. Therefore, $\left(D^m N Q^n\right)[i,j] = v^i q_{r.j-1}$. By the same logic as the proof of Lemma 4, this is maximized along a row when $|i-(j-1)k-1|$ is minimized. □

A.8 Proof of Proposition 5

Due to a standard property of the Hadamard product

$$\boldsymbol{e}^T \left(\left(A\hat{D}_2 R \hat{D}_1 N \Pi\right) \odot U\right) \boldsymbol{e} = tr\left(A\hat{D}_2 R \hat{D}_1 N \Pi U^T\right)$$

By the cyclic property

$$tr\left(\hat{D}_1 N \Pi U^T A \hat{D}_2 R\right)$$

Define

$$M = \hat{D}_1 N \Pi U^T A \hat{D}_2 = \begin{bmatrix} \boldsymbol{m}_1 & \boldsymbol{m}_2 & \boldsymbol{m}_3 & ... \end{bmatrix}$$

Where $\boldsymbol{m}_i$ is the ith column of M. Further define

$$R = \begin{bmatrix} \boldsymbol{r}_1 \\ \boldsymbol{r}_2 \\ \boldsymbol{r}_3 \\ ... \end{bmatrix}$$

Where $\boldsymbol{r}_i$ is the ith row of R. The player's expected utility then becomes

$$\sum_i \boldsymbol{m}_i \cdot \boldsymbol{r}_i$$

Where the dot represents the typical dot product. combined with the fact that R is right stochastic, this give the result. □

A.9 Proof of Remark 4

An optimum exists since the objective is continuous and the space is compact. If one memory state i has a non-zero, non-one probability of being encoded as a memory state j in the optimal re-encoding then re-encoding memory state i as memory state j is weakly optimal. In this case setting the probability to one will not lower payoffs.

A.10 Proof of Proposition 6

Expected utility is given by

$$\max_{R_{t+n+1}} Tr \left(R^{'}_{t+n+1} \hat{D}_{t+n+1} ... R_{t+1} \hat{D}_{t+1} R_t ... R_2 \hat{D}_2 R_1 \hat{D}_1 N \Pi U^T A \hat{D}_{\tau-1} R_{\tau-2} ... \right)$$

For some optimal A, N and other Rs. Define this as

$$\max_{R_{t+1}} Tr \left(R_{t+1} S_1 S_2 M \right)$$

Where $S_1 = M$ or $S_1 = M\hat{D}_t$ (depending on which condition holds), is a stochastic matrix,

$S_2 = ...R_2\hat{D}_2R_1\hat{D}_1N$ is a stochastic matrix, and $\tilde{A} = \Pi U^T A\hat{D}_{\tau-1}R_{\tau-2}...$ is an arbitrary conforming matrix.

Following Leshno and Spector (1992)'s part (a) of the proof of the Blackwell theorem (transposed in this case) we have

$$\max_{R_{t+1}} Tr\left(R_{t+1}S_1S_2\tilde{A}\right) = \max_{R_{t+1}} Tr\left(R_{t+1}GS_1^{'}S_2\tilde{A}\right) \leq \max_{R_{t+1}} Tr\left(R_{t+1}S_1^{'}S_2\tilde{A}\right)$$

Where $S_1 = M'$ or $S_1 = M'\hat{D}_t$ (again depending on which condition holds). □

B Extra Examples

B.1 Simplex Like

Say there is an absorbing memory state A_{12} which can only be reached from encoding state $\tilde{m}_{\theta_1}$ with probability 0.1 or from $\tilde{m}_{\theta_2}$ with a probability 0.2. Further say that the memory system will not return to encoding states once it has left them to avoid repeating paths. If the memory system is Simplex-Like there must also an absorbing memory state A_{ij} for every other pair of encoding states which is reachable from m_{θ_i} with probability 0.1 or from m_{θ_j} with a probability 0.2. This includes $i = 2$ and $j = 1$. The ξ in this only contains single element or no element sequences, so it is easy to write out as $[[0.1], [0.2], [\emptyset], [\emptyset], ...]$.