Supporting Information

Desrochers et al. 10.1073/pnas.1013470107

SI Results

The reinforcement test showed a correlation between changes in pattern and changes in distance. As an extension of this analysis, we examined what effect trial-by-trial changes in the distance that the eyes traveled during the Reward Scan period (cost) would have on the loop sequences. We hypothesized that if cost had an effect on loop sequence performance then a cost benefit would result in the monkey performing a sequence sooner (i.e., after fewer trials), whereas detrimental cost would result in the monkey performing a sequence later (i.e., after many trials). To test this hypothesis, for each trial on which a loop sequence was performed we calculated the intersequence interval (ISI): the number of trials that transpired before the loop sequence was performed again. The ISI for the current trial was then compared to the change in cost from the previous trial.

We found a significant positive correlation between the mean ISI and the mean change in distance (shuffle test, $P \leq 0.05$; Fig. S5, and *SI Methods* for more details). This finding suggests that when the monkey performed a loop sequence (e.g., G9 purple and Y9 yellow; Fig. S5 *A* and *B*), if the monkey's eyes traveled a shorter distance during the Reward Scan on that trial compared with the previous trial (negative cost), then the monkey would repeat that same loop sequence in a fewer number of trials than if there had been a greater cost. This finding held for the pool of loop sequences illustrated in Fig. 2 across both monkeys for both grid sizes (Fig. S5 *C* and *D*). In contrast to the parallel between the monkeys' and the modeled behavior in all previous analyses, this correlation between ISI and cost was not seen in the REINFORCE algorithm-simulated data (Fig. S5*E*). This finding suggests that RL could account for the shifts in the loop sequences as performed by the monkeys, and leaves an avenue for the possible expansion of the RL model.

We used the data produced by the REINFORCE algorithm to validate the reinforcement test used on the monkey data in several ways. First, we used the reinforcement test on the simulated data to confirm the presence of RL in the same manner as in the monkeys (Fig. S7 *Left columns*). The presence of RL was not detected in the simulated data when the reward (negative geometric distance) on each trial was randomly assigned to be a number between 10 and 15, which was approximately the same range as the distances on the REINFORCE simulated trials (Fig. S7 *Right columns*).

Second, to examine how accurate the reinforcement test was, we performed multiple simulations and randomly assigned each simulation to either use REINFORCE or have random reward as described above. Each of the 300 simulations had 100 sessions with 200 trials each session. For each simulation, the reinforcement test was used to calculate the *P* value of the slope of the correlation between the change in distance and the dissimilarity in the change of patterns. For the four- and nine-target simulations, the false-positive rate was less than 0.3% with none of the slopes in the random reward condition simulations having a *P* value of less than 0.01 (Fig. S8). Receiver operating characteristic (ROC) analysis showed there was an improvement in the true positive rate when more sessions were used (200 sessions per simulation), but there were still no false positives, further supporting the use of this test to detect RL.

In this task, measures of distance, reward rate, and number of saccades/fixations were all correlated. Consequently, there existed the possibility that a parameter other than distance would be the best drive for reinforcement learning. Because there was one reward at the end of each trial (reward was constant), reward rate is inversely proportional to the time required to capture the baited target. Similarly, because each saccade was surrounded by two fixations that occupied a larger fraction of time (~150 ms per fixation vs. ~30 ms per saccade), the number of saccades is approximately proportional to the time.

To disambiguate these factors, we used the REINFORCE algorithm and simulated 5,000 sessions of 200 trials each using the reward rate as the reinforcement drive with the goal being to maximize the reward rate. Reward rate was calculated as one over the time to capture the baited target, with the duration of each fixation set to 150 ms and the duration of each saccade set to 30 ms times the geometric distance of the saccade (one unit equals the horizontal or vertical distance between adjacent targets). We found that the session-averaged reward rate, distance, and entropy took a greater number of sessions to reach steady-state than when the distance was used as the reinforcement drive in the simulations (Fig. S9 *A*–*C*). More strikingly, by the 5,000th session the reward rate-based simulation had not converged on the optimal path (Fig. S9 *D* and *E*, black) as had the distance-based simulations (Fig. 6*E*).

The final transition probabilities reached by the reward maximizing REINFORCE simulation, despite not having converged on the most optimal path, could have been close to optimal. To test the performance of the final set of transition probabilities (Fig. S9*E*, black) in comparison with the optimal (Fig. S9*E*, red) or a nearly optimal (Fig. S9*E*, green) in distance path, we generated 50,000 saccade sequences using Monte Carlo simulations and calculated the mean reward rate of each set of transitions. We found that the mean reward rates were nearly equivalent for all three paths: 0.0019422, 0.0019962, and 0.0019986 rewards per ms, respectively (Fig. S9*F*). Thus the reward maximizing simulation did indeed yield a set of transitions that produced a comparable mean reward rate; however, there were too many patterns that also had close to optimal reward rates and the optimal distance path was not converged upon. In contrast, minimizing the distance using REINFORCE had a unique pattern that the monkey also converged on.

For the same three sets of transitions discussed above, the total geometric distance means were 8.0553, 5.8462, and 6.4445, respectively (Fig. S9*G*). The mean simulated number of saccades to complete the trial was similar with the final reward maximizing simulation transitions being the greatest and the optimal, and nearly optimal distance paths had a mean of fewer saccades (Fig. S9*H*). Therefore, to simulate this task, minimizing distance more closely approximated the monkeys' behavior and converged on the optimal policy, whereas maximizing reward rate did not.

SI Methods

Behavioral Data Acquisition. Two adult female monkeys (*macacca mulata*) were studied (~5.9 kg each). Eye position was monitored by infrared eye tracking (500 Hz; SR Research Ltd.) and recorded with a Cheetah Data Acquisition system (2 KHz; Neuralynx, Inc.). Custom behavior control software was designed in Delphi III. The monkey was seated ~50 cm from the LCD screen, and a "hot" mirror that passed visual light and reflected infrared light to the camera above was placed at a 45° angle in front of the monkey. Before behavioral data acquisition, a head post and recording chamber were surgically implanted under pentobarbital anesthesia using sterile methods according to National Institutes of Health guidelines and as approved by Massachusetts Institute of Technology's Committee on Animal Care.

Behavioral Procedures. Each free-viewing scan trial consisted of task epochs as shown in Fig. 1, with slight differences between the two monkeys. The Reward Delay was 400–800 ms (G) or 600–1,200 ms (Y). Reward Time was 200 ms of liquid juice (G) or 250 ms of food mush (Y), and the ITI was 2 s (G) or 3 s (Y). The gray target grids had either 49 (1.6° diameter, 4.1° spacing, monkey G) or 36 (2.0° diameter, 5.4° spacing, monkey Y) targets. The green target grid replaced the inner portion of the gray grid leaving a perimeter of gray targets and consisted of either four or nine targets (diameters: 4.0°, 2.7°; center-to-center spacing: 13.9°, 7.7°, respectively), depending on task acquisition and performance. Both grids spanned the central 17.6° of the screen, and the sum of the total area of green was equal.

Once the green target grid appeared at the end of the Start Delay, the monkey had 5 s to enter the green grid space, although the monkey's eye position was usually already within that space or became so on the next saccade. On the rare occasion when the monkey's eye position did not enter the green grid space, the trial was aborted. During the Reward Scan, the start of which was not signaled to the monkey, the pseudorandomly chosen baited target could be captured by the monkey's gaze either fixating on or saccading through the target. The only constraint placed on the monkey's eye position throughout the Total Scan Time was that it had to remain within the area defined by the green target grid when it was displayed. Exiting this grid resulted in the trial being aborted by extinguishing the green grid and proceeding directly to the ITI with no reward delivered.

The first session of task acquisition required an initial estimate of the eye position in relation to the targets. The four-target scan grid was displayed and a treat (e.g., a raisin) was placed in front of one of the targets. The gain and offset of the eye position signal was adjusted so that when the monkey's eye position was near the treat/target, it was rewarded via the tube in front of its mouth. This procedure was repeated for all four targets. Subsequent recording sessions also included a short period of calibration using the four-target task, but without the assistance of treats.

Throughout task acquisition, three parameters were manipulated to shape the monkey's behavior: (*i*) rewarding any target that was captured after the Delay Scan (Fig. S1*B*); (*ii*) adjusting the size of the window around each target that would trigger capture (Fig. S1*C*); and (*iii*) adjusting the duration of the Delay Scan (Fig. S1*D*). This third parameter was the main one used for shaping. During initial task acquisition, the Delay Scan was negligible (1–2 ms) but was increased in increments of ~50 ms (e.g., 50–100 ms, 100–200 ms) as guided by the monkey's task performance. In general, the monkey's performance was considered suitable for an increase in Delay Scan time if in the two preceding groups of 50 trials the monkey had attained 80% rewarded trials or better. The observation that the monkeys continued to move their eyes during the Delay Scan when they were not required to do so (see Fig. S2 *G–I*) could be a result of this shaping procedure. However, it could also indicate that the automatic execution of saccade sequences in a habited manner was "easier" than timing when the baited target would become available for capture.

Each session consisting of ~1,000 rewarded trials was divided into the following blocks: rest, calibration, five scan task blocks, rest, five scan task blocks, and rest. In the rest block the monkey sat passively in front of a black screen for a period approximately equal to 40 trials (~5 s each). The calibration block was typically 5–15 trials and was occasionally repeated later in the task session if drift of the eye position was suspected. Each scan block consisted of ~100 rewarded trials of a single grid size with each target chosen to be baited an approximately equal number of times. Monkeys first acquired the four-target task. When more than one grid size was used, monkey G had one block per half-session of the smaller grid size(s), and the remaining blocks were the largest grid size (e.g., 100 four-target trials, 400 nine-target trials in a half-session). This structure was simplified for monkey Y so that all scan blocks would be the same target grid size with the exception of one session per week where half the scan blocks were four-target and the other half were nine-target (e.g., 250 four-target trials, 250 nine-target trials in a half-session). Grid sizes with more than nine targets were attempted with monkey G (16 and 25 targets), but were not included for analysis or performed with monkey Y, due to the difficulty the monkeys had in scanning them.

Data Analysis. One Y4 and one Y9 session were excluded due to data loss. Session blocks with fewer than 55% rewarded trials were only included if performance on other blocks indicated the monkey was sufficiently motivated to perform the task, and session blocks with fewer than 40 rewarded trials were not included in analyses (~4% excluded overall). All eye movement analysis was done in Matlab.

Horizontal and vertical eye position traces were postprocessed offline in preparation for data analysis as follows. Traces were filtered to remove and interpolate between short (<2 ms), high-velocity events that resulted from the eye-tracking camera momentarily not being able to track the eye and then smoothed using a 33-sample wide Hanning window. Slight calibration adjustments were then made by hand to ensure the eye traces were in proper register with the displayed targets. Eye traces were then separated into fixations, saccades, and blinks. Velocity thresholds for saccades and blinks were automatically set on a session-by-session basis according to the distribution of velocities for the entire session. Position thresholds for blinks were also found automatically by determining the maximum values of the horizontal and vertical eye positions per session. Blinks were then parsed by determining spans of time >500 ms that exceeded the blink velocity and/or eye position thresholds. A small number of sessions were found to have oscillatory noise; if this was the case, blinks were removed from the eye traces; the resulting gap was filled by interpolation and the resulting trace was low-pass filtered using a Butterworth filter. Saccades were parsed by identifying nonblink velocity threshold crossings and then fitting them with a Gaussian or sum of Gaussians, depending on the complexity of the velocity profile. All times that were not categorized as a blink or saccade at the end of this process were marked as a fixation. Fixation and saccade event markers were then "cleaned" to prevent impossibilities, such as durations that were too short or too long or that included many overlapping events in a short period (>30 events per sec in a 200-ms sliding window). Eye movements were tabulated trial-by-trial for import into a database (Microsoft Access or Post GRE SQL) and subsequent data analyses.

The most frequent eye movement paths or "loop" sequences were assembled by identifying the top 20 unique sequences of five fixations with greater than 20 occurrences and an incidence of greater than 1% in a single session. Sequences of length five were used after examining all sequence lengths and obtaining similar results; there were a manageable number of five-fixation sequences. These were pooled and inspected for overlapping fixation sequences that completed a full loop (began and ended on the same target). The percentage of trials that contained any of the permutations of the full loop (regardless of start or stop position and, if applicable, regardless of whether a middle target was stopped on or passed through) was tallied.

The session at which behavioral measures reached asymptote was determined as follows. Starting with all sessions and then one-by-one subtracting beginning sessions (e.g., sessions 1–60, 2–60, 3–60, etc.), data were fitted with regression lines. The slope of each resulting line was tested with a *t* test to determine whether it was significantly ($P < 0.01$) different from zero. The data were said to have reached asymptote when the slope of the resulting line fit was not significantly different from zero.

The fraction of trial fixations that were members of the loop sequences was calculated by first determining whether each fixation in each trial was a member of any of the loop sequences for that condition, as shown in Fig. 2. Then the fraction of fixations that were members of the loop sequences over all fixations in that trial was calculated. This fraction was averaged, session-by-session, for all rewarded trials.

Transition probabilities used for nonnegative matrix factorization (NMF) were calculated as the probability of a saccade from one target to the next (given fixation on the start target). The number of factors for NMF was determined by computing the rms residual of the factorization using up to 10 (four-target) or 20 (nine-target) factors and determining the number of factors necessary for the residuals to complete an initial drop (an "elbow method"). Two NMF decomposition algorithms were applied, an alternating least-squares algorithm and Seung's multiplicative update algorithm (1), and the result with smallest residual was chosen (as this best optimizes the NMF objective function). The component factors were normalized.

The statistical dispersion of each NMF factor was assessed by multiplying the absolute value of distance from the median by the height (value) at each session. The value of each factor at each session was then shuffled and the same statistic computed again. This process was repeated 10,000 times, and the *P* value was the fraction of runs with a statistic less than that of the nonshuffled factor.

The most optimal deterministic patterns were determined using an exhaustive search algorithm with the following three criteria: the solution (*i*) must have the shortest path length; (*ii*) must cover all targets in the grid; (*iii*) must start and end on the same target to form a closed "loop."

To reduce variability in the distance calculation due to behavioral shaping for the reinforcement test and intersequence interval (ISI) analyses, we calculated the geometric distance the monkey's eyes traveled during the Total Scan Time using only those trials in which the Delay Scan was >1 s and the Reward Scan had more than one saccade. After this initial elimination of trials, analysis was restricted to those trials that remained consecutive. In addition, saccades (distance traveled) in error trials were added to the next rewarded trial, as there was no reward delivered during error trials.

Formally, the reinforcement test was calculated as follows. Reward was calculated as the negative of total distance. The delta distance is defined as

$$D_k = d_k - d_{k-1}, \qquad \textbf{[S1]}$$

where d_k is the total geometric distance of the saccades in the *k*th trial. Positive D_k is punishment, and negative D_k is reward. To measure the difference in saccade patterns, we compute the transition probabilities P_k between the targets in the *k*th trial. Note this will be a vector with N^2 components ($N = 4$ or 9 targets). The change in the scan pattern is then defined as

$$\Delta(k, k-1) = P(k) - P(k-1). \qquad \textbf{[S2]}$$

We then computed the delta pattern dissimilarity using the cosine distance measure:

$$S_k = \frac{\Delta(k, k-1)\cdot\Delta(k+1, k)}{|\Delta(k, k-1)||\Delta(k+1, k)|}. \qquad \textbf{[S3]}$$

Here, $|\Delta(k, k-1)|$ and $|\Delta(k+1, k)|$ are the lengths of the vectors. To detect the correlation between D_k and S_k, we selected all trials with $|D_k| < D_{max}$, where D_{max} is equal to the median of $|D_k|$ plus 3× the SD. This selection eliminated ~2–3% of the points in D_k at the extremes. The resulting selected trials were then pooled into 10 equal-size bins, and the median change in distance (*D*) and pattern dissimilarity (*S*) was calculated for each bin. A regression line was fit to those data. To determine if the slope of the line was significant, we randomly shuffled *S* 500 times and computed the slope of the resulting regression line after binning by the same procedure. The *P* value is given by the fraction of times the actual slope is less than that of one of the shuffles.

The correlation between the change in distance (*D*) and the ISI was calculated in the same manner as for the reinforcement test. The data were pooled into 10 equal-size bins and the mean change in distance and ISI was calculated for each bin. A regression line was fit to those data. To determine if the slope of the line was significant, we randomly shuffled ISI 500 times and computed the slope of the resulting regression line after binning by the same procedure. The *P* value is given by the fraction of times the actual slope is less than that of one of the shuffles.

We constructed the REINFORCE algorithm as follows. The agent followed a Markovian decision mechanism to generate saccades. From target *j*, the probability of a saccade to target *i* was p_{ij}. Here $i, j = 1, \ldots, N$, where $N = 4$ or 9 targets. A "start target" was designated as $j = 0$ that corresponded to the initial state of the agent outside of the targets. The transition probabilities were determined from the values m_{ij} associated with the transition from *j* to *i*. To determine the probabilities from the values in a manner that would balance discouraging the appearance of suboptimal stereotypical patterns too early, and allowing enough exploration without slow exploitation, we used the following method:

$$p_{ij} = \frac{m_{ij}^4}{\sum_j m_{ij}^4}. \qquad \textbf{[S4]}$$

We generated exploration in m_{ij} through sampling a Gaussian distribution:

$$p(m_{ij}) = N(M_{ij}, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}}\, e^{\frac{\left(m_{ij}-M_{ij}\right)^2}{2\sigma^2}}. \qquad \textbf{[S5]}$$

Here, M_{ij} is the mean of the action values, and σ is the SD. To keep the size of the exploration comparable to the mean, we scaled uniformly at each trial:

$$M_{ij} \rightarrow (1-\delta)\, M_{ij}. \qquad \textbf{[S6]}$$

Note that a uniform scaling of M_{ij} does not change the averaged transition probabilities. The constant reduction of M_{ij} balanced the growth of M_{ij} through learning, which kept exploration greater than a minimal level. The learning rule used so that reward would influence how the mean values changed for the exploit step was as follows:

$$M_{ij} \rightarrow M_{ij} + \alpha\,(r(t) - \bar{r})\left(m_{ij} - M_{ij}\right), \qquad \textbf{[S7]}$$

where $r(t)$ is the reward at time step *t*, $\bar{r}$ is the averaged reward of past trials, and α is the learning rate. The averaged reward of past trials can be estimated using:

$$\bar{r}\ (t) = \gamma r(t-1) + (1-\gamma)\bar{r}(t-1). \qquad \textbf{[S8]}$$

This is a discrete version of the differential equation:

$$\frac{d\bar{r}(t)}{dt} = \gamma\left(r(t) - \bar{r}(t)\right). \qquad \textbf{[S9]}$$

The above learning mechanism can be mathematically derived from the REINFORCE mechanisms proposed by Williams (2).

Our agent had random action values to start with. At each trial, new action values were sampled from the Gaussian distribution and these values determined the transition probabilities from which the sequence of saccades was generated following the same task constraints as the monkeys (1- to 2-s Delay Scan and a randomly baited target). The parameters used in the simulations were: $\sigma = 0.1$; $\gamma = 0.5$; $\delta = 10^{-5}$; $\alpha = 10^{-2}$ for four targets and $\alpha = 2 \times 10^{-3}$ for nine targets. These parameters were selected to ensure fast convergence of the saccade patterns to the optimal patterns. Additionally, at each saccade there was a 1% chance that the simulation would make an "error" and abort the trial (11–13% of trials, similar to error rate of monkeys). This allowed the termination of saccade patterns which did not cover all the targets. The distance or time spent in an error trial was added to the next trial's values.

1. Lee DD, Seung HS (1999) Learning the parts of objects by non-negative matrix factorization. *Nature* 401:788–791.

2. Williams RJ (1992) Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach Learn* 8:229–256.

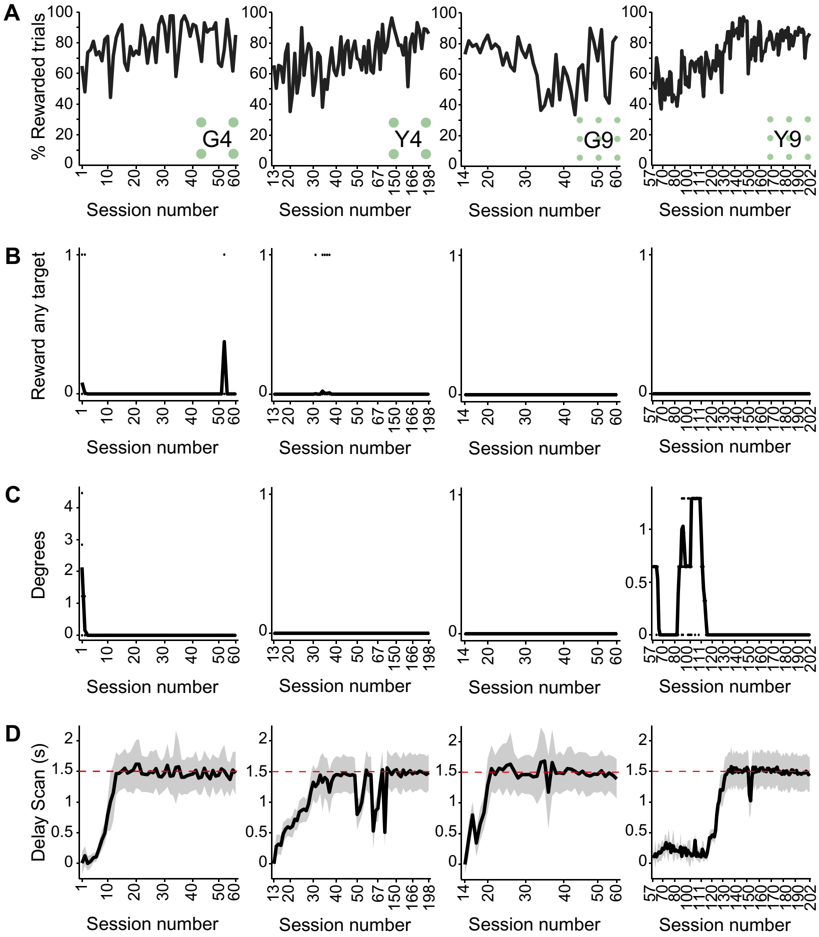

Fig. S1. Task performance and shaping parameters. All rows show the monkeys and the conditions in the following order: G4 (monkey G in four-target task), G9, Y4, and Y9. (*A*) Percent of rewarded trials per session. (*B* and *C*) Dots indicate values of parameters in each session; lines indicate the average of the values. (*B*) Indicates for each session the presence (1) or absence (0) of the parameter rewarding any target entered after the Delay Scan. Note this parameter was not used in G9 or Y9. (*C*) In each session, the number of degrees the radius of the acceptable window around the target was greater than the radius of the target. Note this parameter was not used in G9 or Y4. (*D*) Mean ± SD. Delay scan time for each session. Red dashed line for all plots indicates the mean of 1.5 s that is reached when the monkey completes task acquisition and performs the full 1- to 2-s Delay Scan.

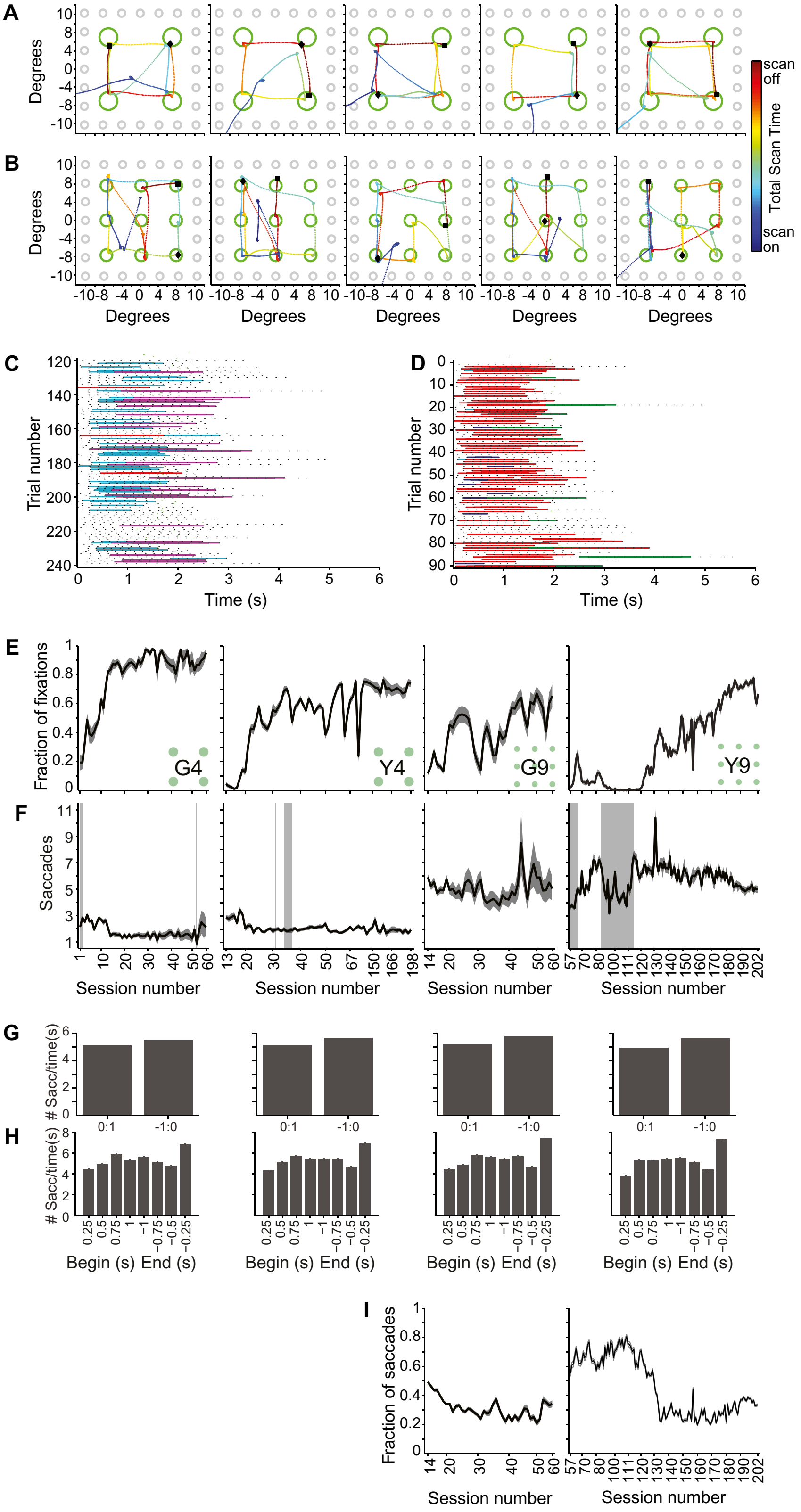

Fig. S2. Example trials and scan pattern statistics. (*A* and *B*) Eye movements of monkey G for five sequential trials during the Total Scan Time. Time is represented by color from dark blue (green targets on) to dark red (green targets off). (*A*) Session 12, four-target task. (*B*) Session 21, nine-target task. (*C* and *D*) Fixation rasters with loop sequences highlighted. Each horizontal line represents the Total Scan Time for each rewarded trial for G9 trials during session 21 (*C*) and session 60 (*D*). Time 0 is the onset of the green target grid. Black dots indicate fixation onset times. Horizontal colored lines indicate fixations that are members of the loop sequences shown in Fig. 2, with the same color code. (*E*–*I*) Monkey and task condition for each row as indicated by cartoon in the lower right of row *E*. Dark gray horizontal shading indicates approximate confidence limits where applicable (±1.96 × SEM). (*E*) Mean fraction of fixations per trial

Legend continued on following page

during the Total Scan Time that are members of any of the loop sequences as diagrammed in Fig. 2 for each monkey in each task condition (e.g., Y4 shows fraction of fixations per trial that are members of any of the three paths diagrammed in Fig. 2*B*). (*F*) Mean number of saccades per Reward Scan period. Gray shaded vertical bars indicate sessions containing shaping periods when the task was made easier for the monkey (see *Methods* and Fig. S1). (*G*) Mean number of saccades per second (±1.96 × SEM; note error bars are too small to be seen) in the first second after the green target grid turns on (begin range 0:1 s) and the last second before the baited target is captured and the green target grid turns off (end range −1:0 s). (*H*) Same as in *G* with saccade rate in the first and last seconds shown in 0.25-s bins with greater bin edge listed for each bin, e.g., begin range 0:0.25 s, 0.25:0.5 s, end range −0.5: −0.25 s, −0.25:0 s, etc. Note the last bin (−0.25:0 s) for each monkey and task condition is likely the largest because by definition the green targets will turn off when the baited target is captured with a saccade, thus making the minimum number of saccades in that bin 1 instead of 0. (*I*) Mean fraction of saccades per trial that pass through intervening targets.

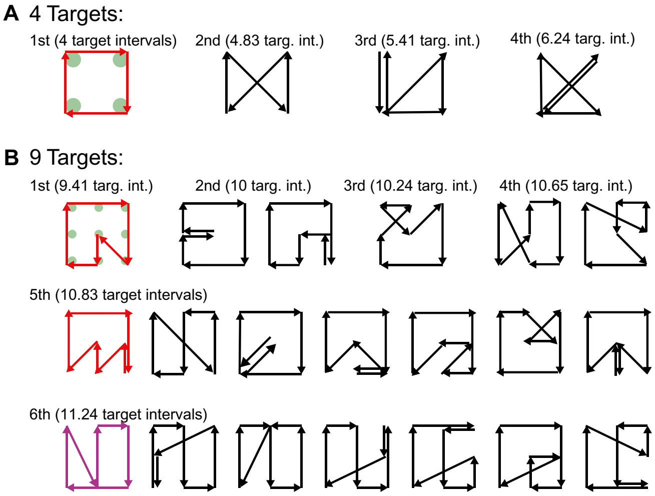

Fig. S3. Monkeys converge on efficient deterministic patterns. The most efficient deterministic patterns for the four-target (*A*) and nine-target (*B*) tasks as computed using an exhaustive search algorithm are shown. Total geometric distance to cover all of the targets using the depicted path is noted. Only one of many possible equivalent rotations and reflections is shown for each path. Paths the monkeys performed are colored red or purple to correspond with Fig. 2.

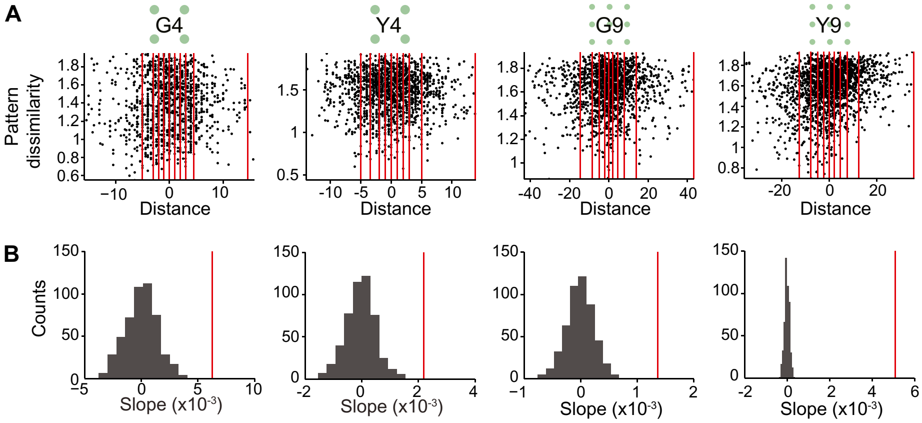

Fig. S4. Trial-by-trial reinforcement test shows correlation between cost and change in pattern. Each column corresponds to the monkey and task conditions as depicted above row *A*. (*A*) The overall distribution of change in geometric distance and dissimilarity of the change in pattern, where pattern is the set of transition probabilities representing a single trial. Each point represents one trial: G4: *n* = 6,109 trials; Y4: *n* = 25,113 trials; G9: *n* = 5,912; and Y9: *n* = 54,214. (*B*) Histogram of slopes resulting from shuffling the dissimilarity of change in the patterns 500 times and computing the resulting slope for each. Actual slope indicated by red line and depicted in Fig. 5. All shuffled slopes were found to be less than the actual slope, indicating a significance of $P < 0.002$.

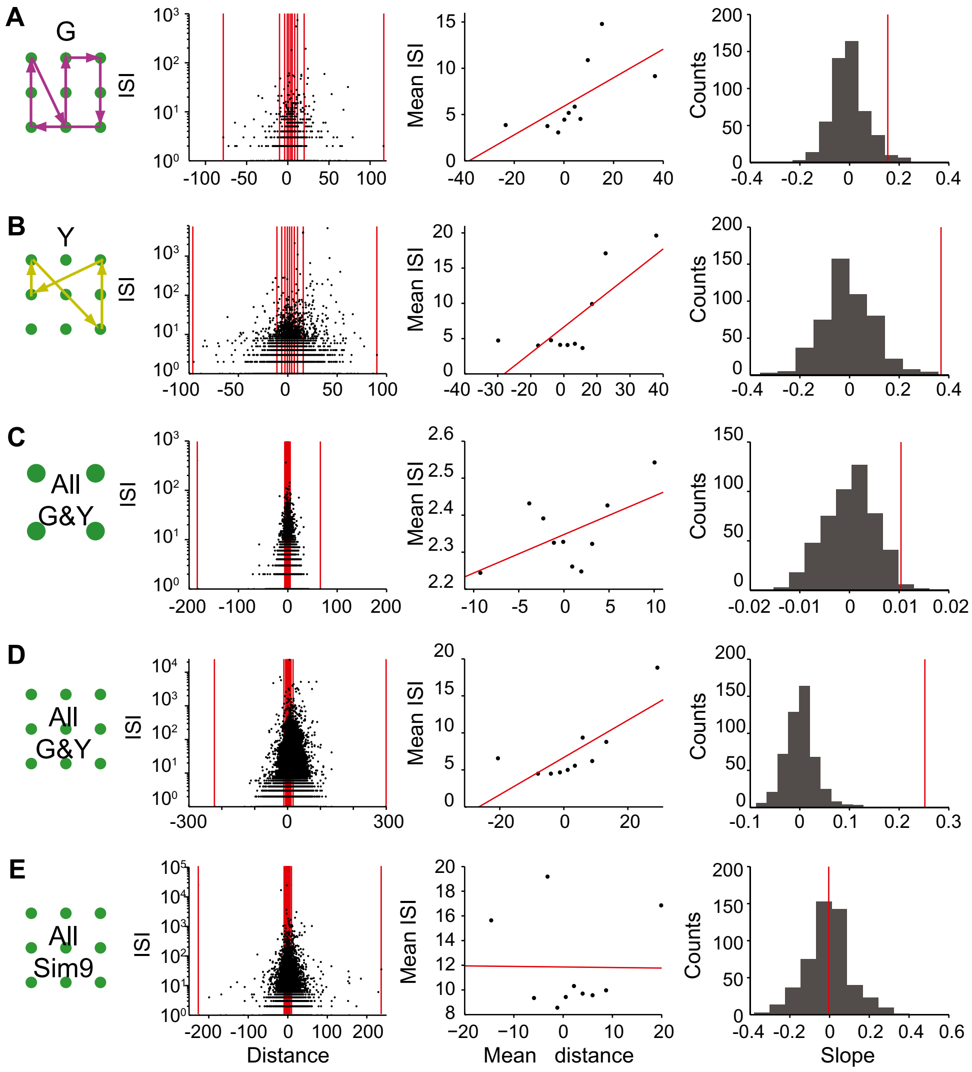

Fig. S5. Correlation between trial-by-trial change in distance and intersequence interval (ISI). All rows contain the same three elements as follows. (*Left*) Overall distribution of change in geometric distance and ISI across all trials containing the pattern(s). Each data point is a trial with the number of trials (n) listed below. (*Center*) Mean and line fit of each of 10 bins each containing the same number of trials (bin edges indicated by red lines in left column). Linear correlation coefficient (R) and correlation p value listed below. (*Right*) Histogram of slopes resulting from shuffling the ISI and change in distance 500 times and computing the resulting slope for each. Actual slope illustrated in center column indicated by red line. P value listed is fraction of shuffled slopes greater than actual. (*A*) ISI correlation for an example G9 path: n = 878; R = 0.625, p = 0.053, slope = 0.15; P = 0.03. (*B*) ISI correlation for an example Y9 path: n = 6,702; R = 0.787, p = 0.007, slope = 0.37; P < 0.002. (*C*) Pool of all trials containing any of the four-target paths diagramed in Fig. 2 *A* and *B*: n = 35,350; R = 0.564, p = 0.089, slope = 0.01; p = 0.01. (*D*) Pool of all trials containing any of the nine-target paths diagrammed in Fig. 2 *C* and *D*: n = 76,885; R = 0.766, p = 0.010, slope = 0.25; P < 0.002. (*E*) Pool of all trials containing any of the simulated nine-target task (Sim9) loop sequences; note there is no correlation: n = 29,953; R = −0.011, p = 0.976, slope = −0.005, P = 0.49.

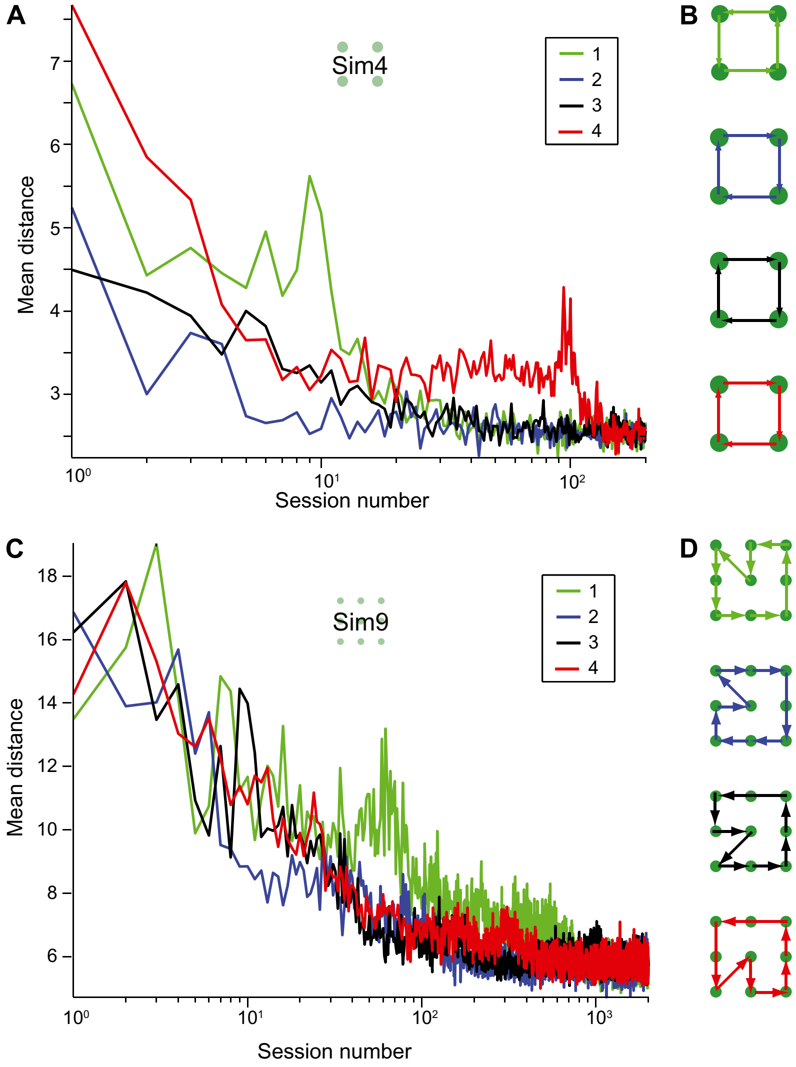

Fig. S6. Simulations have different learning rates but converge on the same paths. (*A*) Mean geometric distance over 200 sessions, each containing 200 trials, of four REINFORCE algorithm simulations of the four-target task with corresponding final paths shown in *B*. (*C*) Mean geometric distance over 2,000 sessions containing 200 trials each of four REINFORCE algorithm simulations of the nine-target task with corresponding final paths shown in *D*.

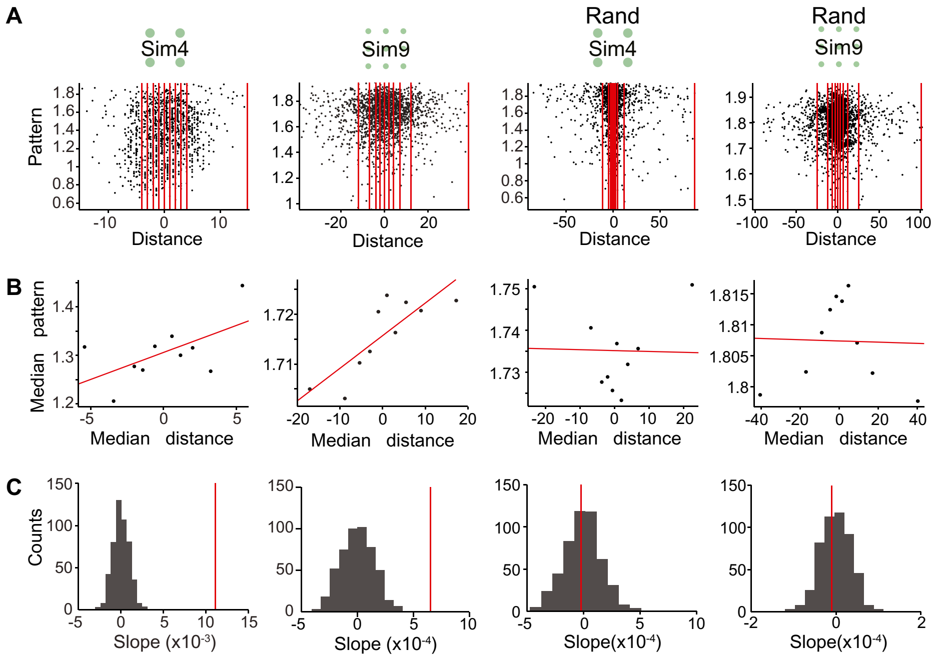

Fig. S7. Trial-by-trial reinforcement test shows correlation between cost and change in pattern for REINFORCE but not random simulations. Each column corresponds to the simulation and task conditions as depicted above row *A*: four-target simulation (Sim4), nine-target simulation (Sim9), four-target simulation with random reward (Rand Sim4), and nine-target simulation with random reward (Rand Sim9). (*A*) Overall distribution of change in geometric distance and dissimilarity of the change in pattern, where pattern is the set of transition probabilities representing a trial, across all trials (*N*): Sim4, n = 16,715; Sim9, n = 16,732; Rand Sim4, n = 16,144; Rand Sim9, n = 16,350. (*B*) Median change in geometric distance and dissimilarity of pattern change for each of 10 bins containing equal numbers of trials. Bin edges indicated by red lines in *A*. Slope of line fit to the 10 points shown with correlation coefficient (*R*) and *p* value listed: Sim4, $R = 0.580$, $p = 0.079$, slope = 0.01; Sim9, $R = 0.825$, $p = 0.003$, slope = 0.0007; Rand Sim4, $R = -0.025$, $p = 0.946$, slope = -2.1×10^{-5}; Rand Sim9, $R = -0.031$, $p = 0.933$, slope = -9.9×10^{-6}. (*C*) Histogram of slopes resulting from shuffling the dissimilarity and change in distance 500 times and computing the resulting slope for each. Actual slope as found in *B* indicated by red line. *P* value listed is fraction of shuffled slopes greater than actual. Note left two columns (REINFORCE simulations) show significant slope ($P < 0.002$) and right two columns do not (random reward simulations, $P = 0.58$ and $P = 0.60$, respectively).

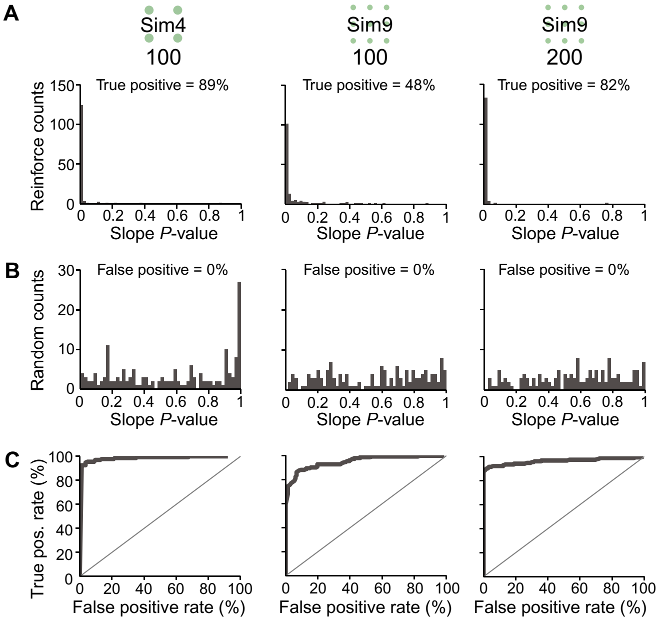

Fig. S8. Reinforcement test detects RL with almost no false positives. Simulation and task condition depicted above row *A*: Sim4 100, 300 simulations of 100 sessions each; Sim9 100, 300 simulations of 100 sessions each; Sim9 200, 300 simulations of 200 sessions each. All simulated sessions contained 200 trials. Each simulation was randomly assigned to the REINFORCE algorithm or random reward algorithm. (*A*) Histogram of the *P* values of the slopes obtained using the reinforcement test on each of the simulations assigned to the REINFORCE algorithm. True positive rates noted are the percentage of *P* value counts where $P < 0.01$. (*B*) Histogram of the *P* values of the slopes obtained using the reinforcement test on each of the simulations assigned to the random reward algorithm. False positive rates noted are the percentage of *P* value counts where $P < 0.01$. (*C*) Receiver operating characteristic (ROC) curve for the results of the 300 simulations in each condition.

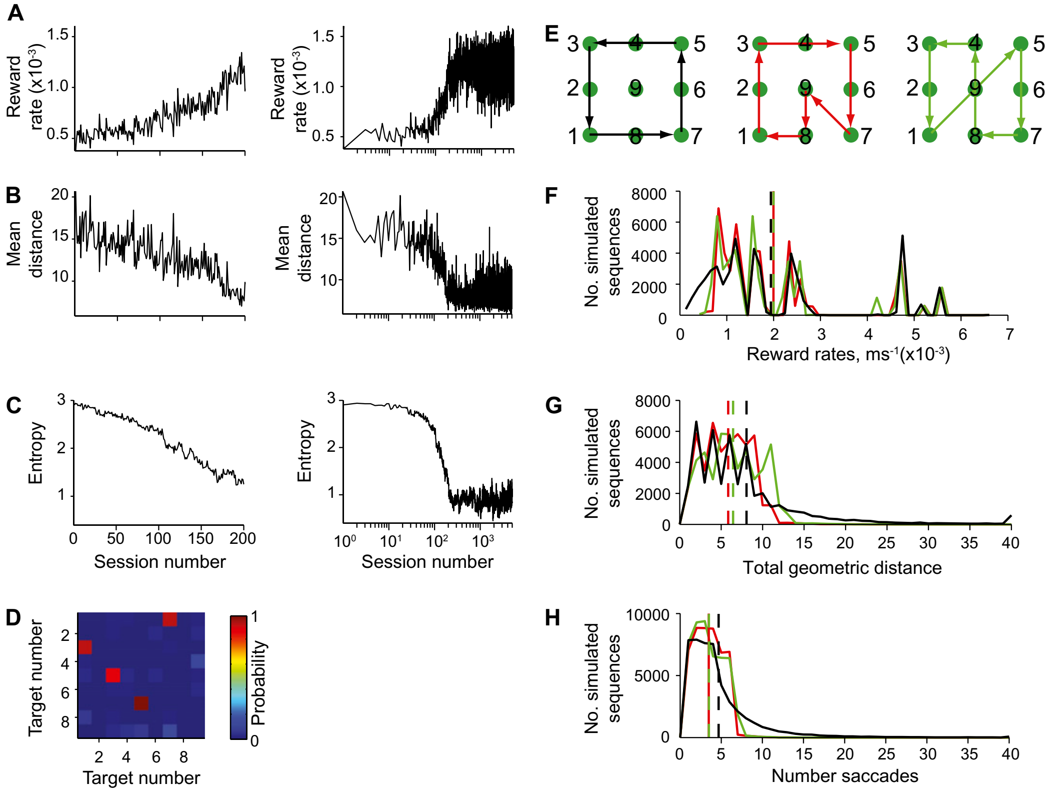

Fig. S9. REINFORCE simulation using reward rate instead of distance does not converge on optimal path. Simulation of 5,000 nine-target sessions, 200 trials each using reward rate (1/trial time with fixations = 150 ms and unit distance saccades = 30 ms) as the reward signal. (*A*–*C*) Left column is the first 200 simulated sessions and right column is all 5,000 simulated sessions. (*A*) Mean reward rate per session. (*B*) Mean geometric distance per session. (*C*) Entropy per session. (*D*) Final transition probabilities in the 5,000th session. Note the very low probability of visiting the center target. (*E*) Black: most probable path corresponding to *D*. Red: optimal distance path (Fig. S3*B*). Green: near-optimal distance path (Fig. S3*B*) with the same number of fixations as the optimal path. (*F*–*H*) Histogram of reward rates generated by 50,000 saccade sequences in Monte Carlo simulations of the task using the paths depicted in *E*. Vertical dashed lines indicate the mean of each distribution. (*F*) Histogram of reward rates. Note the mean reward rate is nearly identical for all three paths. (*G*) Histogram of total geometric distance. (*H*) Histogram of the total number of saccades.