SUPPLEMENTARY MATERIALS

Decoding Procedure. We applied a decoding procedure to further describe the information about the impending movement(s) that is encoded by the planning activity of the whole population. A one-nearest-neighbor classification algorithm (Duda et al., 2001) classified every given test trial by comparing the neural response of the whole population during this test trial with the population responses during other trials. The decoding was done for the PRR populations in each monkey separately. The datasets that were used for the training of the algorithm as well as for the decoding consisted of 70 neurons for monkey C and of 42 neurons for monkey Z. Since the number of trials that had been recorded in the various neurons under each condition was not always the same, we included only the first five recorded trials per condition in the dataset. Each of these five trials was then classified (as a test trial) based on the similarity of its neural signature in the population and the activation vectors in all other (four) trials used as a training set ('leave-one-out decoding' procedure). This procedure was iterated 500 times. The test trial was assigned to the same class as the training trials with the smallest Euclidian distance (i.e., its 'nearest neighbor') within the n-dimensional activity space (n corresponds to the number of cells). The decoding performance was defined as the percentage of correctly classified test trials. We combined trials from single- and double-reach blocks and used a general classification system to predict the direction of the upcoming reach as well as the task type, i.e. whether the movement is going to be a single- or double reach ('full decode'). For a subset of 24 cells in one monkey that had been recorded in all three variants of the reaching task (single-reach, rapid double-reach and slow double-reach) we used the same classification procedure to decode the direction and the type of task with the three levels: SR, rDR and sDR (see Supplementary Material).

Decoding of movement directions and movement types. We applied a nearest-neighbor classification algorithm to better describe how much information about the type of movement is provided by the planning activity of the recorded PRR population before the movement starts. In our full-decode procedure we used the planning activity during the late memory period in order to classify test trials according to (a) the direction, in which the planned movement or movement sequence will lead (direction), and (b) whether the monkey prepares for a double- or single-reach (type of task). The decoding performance, i.e. the probability to successfully predict the actual type and direction of the upcoming action from the preceding planning activity, can shed some more light on how the reach plans are organized in the recorded parietal population. Overall the decoding performance was good with on average 61% correct classifications of test trials (55% in monkey C and 67% in monkey Z as compared to the chance level of 6.7%) as can be seen from the successful classifications along the diagonals in Figure S1. The applied nearest-neighbor algorithm was slightly more successful in categorizing single-reaches compared to double-reach sequences (70% versus 54% correct classifications). A closer look at the SR-decoding (lower left quadrants in Fig. S1 A and B) shows a slightly better decoding performance for goals in the contralateral hemifield (directions 1, 2, and 8 in monkey C compared to the ipsilateral locations 4, 5, and 6 and vice versa in monkey Z). This fits to the observation that 68% of all recorded cells (46 in monkey C and 30 in monkey Z) had their response fields in the contralateral hemifield.

Analysis of Misclassifications in the Nearest-Neighbor decoding:

The applied nearest-neighbor algorithm was not only successful in predicting the direction of the upcoming movement, but also whether it was going to be a single- or a rapid or slow double reach. The applied algorithm rarely mistook the activation vector of a given double-reach test trial as indicating a single reach, nor vice versa (i.e. it could reliably classify whether the test trial belonged to a single- or double-reach block). This speaks in favor of a representation of reach sequences that is clearly different from that of single reaches to the respective goal locations. Most of the rare misclassifications (most of them in monkey C) fit what we expect when applying a nearest-neighbor algorithm to a population, which similarly represents immediate and subsequent goals. They provide interesting insights on how separable the representations of the two reach types towards the various target locations are. Overall, three main types of errors occurred. First, misclassifications of the movement type (single- versus double-reach, 'task error') rarely happened, either for DR test trials or for SR test trials. The algorithm never misinterpreted a given planning activity for a reach sequence as the population signature of an upcoming single reach (nor vice versa), even though both of them share the same first goal location by experimental design (see upper left and lower right quadrants in Figures S1 A and B). Second, for some directions the test trials were misclassified into a wrong but nearby class of direction (see the light grey patches that cluster around the dark diagonal in the lower left quadrant of Figure S1 A, 'direction error'). Most interestingly, these direction errors occurred mainly in the decoding of single reaches, and less for double-reach sequences. Third, in some DR test trials the classes that share a common first or second goal were mistaken for each other ('rank error'). This happened for example in sequence 5/2, which leads first to goal 5 and then to goal number 2 (see dashed arrows in the circular arrangement of Figures S1 A and B). The algorithm mistakenly classified neural planning activity for this sequence with a higher probability as belonging to sequence number 2/7, most likely due to difficulties in distinguishing the neural representations of the first and second goal (similar in sequence 6/3, which is sometimes confused with 3/8).

The 'direction-errors', which occurred mainly in one monkey (monkey C) in single-reach classifications, were substituted by 'rank-errors' in the decoding of double-reach test-trials. The reason for this is that activation patterns for sequences that share a common first or second goal are in some cases more similar to each other than sequences towards slightly different directions.

The successful decoding and the different error patterns in the decoding of single- versus double-reaches provide additional evidence for the assumption that the second goal is similarly activated as the first one.

Decoding of direction-, task and speed of the upcoming movement:

We applied the same nearest-neighbor algorithm also to the subset of cells that were tested under slow versus rapid sequence production. The activity of all recorded cells during the late memory period was used to predict whether the action that is about to be programmed would be a single movement or a fast or a slow sequence of movements.

Figure S2 summarizes the decoding of movement directions and movement types in those 24 cells of monkey C. Despite of considerable confusion about the direction of planned double-reaches one interesting finding is that the algorithm only rarely misclassified test trials with slow double-reaches as rapid sequences. This speaks in favor of clearly distinguishable planning activities conveying the behavioral context for differently speeded sequences in the population.

FIGURE LEGENDS

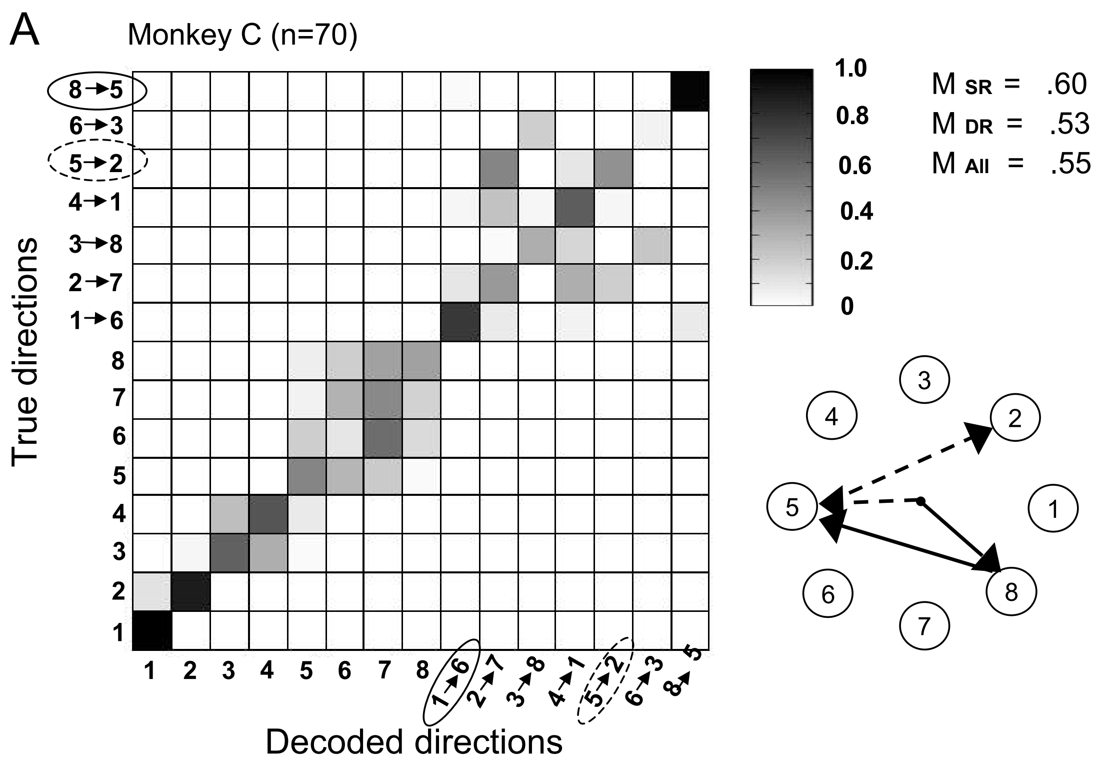
Figure S1: Confusion matrices resulting from 'leave-one-out' decoding procedures using a 'nearest-neighbor' algorithm

The true reach goal directions are aligned on the ordinate. Numbers 1-8, correspond to the respective positions in the circular array to the right. On position 9 to 15 are the respective double-reach sequences. Here, the numbers indicate the location of the respective first and second reach goal. Numbers of the decoded directions are given on the abscissa. The grey-scale indicates the probability, with which the algorithm predicts the goal location(s) in a given test trial. Perfect decoding, i.e. a perfect prediction of the goal location(s) from the planning activity in the population, would result in values close to 1 (dark) on the diagonal and values of 0 (white) at all remaining positions. **(A)** Classification of target locations in the single- and double-reach task based on the spiking activity in 70 cells from monkey C during the late memory period. The classification results are averaged over the predictions in 500 iterations. Note that monkey C could not perform sequence 7/4 because his arm occluded position 7 during the cueing period. Note further that the second goal is always separated by 3 positions (135 deg) in clockwise direction. The reach sequences to the right give an example of a double-reach that was perfectly decoded from the planning activity (sequence 8/5, solid arrows) and for a sequence where the decoding failed (sequence 5/2, dashed arrows), probably due to confusion of 1st and 2nd goal. **(B)** Decoding of single- and double-reaches based on a population of 24 cells in monkey Z during the late memory activity (monkey Z could not perform sequence 2/7).

Figure S2: Decoding of the direction and task variant based on a subset of 24 cells in monkey C that were recorded under all three task conditions. Decoding performance is lower because the population is smaller. Interestingly, the algorithm almost never misclassified a slow double-reach as a rapid one, nor vice versa.

Figure S3: Average activity of all 36 cells that were recorded in the slow version of the double-reach task. The activity is shown time-locked to the end of the first reach component, i.e. the beginning of the 2nd delay interval of 600-750 ms between the two reaches. During this 2nd delay period the cells better represented the second goal, which was still intended (red line), rather than the first goal, which had been accomplished already (blue line).

Figure S1


A
Monkey C (n=70)
True directions
Decoded directions
8→5
6→3
5→2
4→1
3→8
2→7
1→6
1.0
0.8
0.6
0.4
0.2
0
M SR = .60
M DR = .53
M All = .55

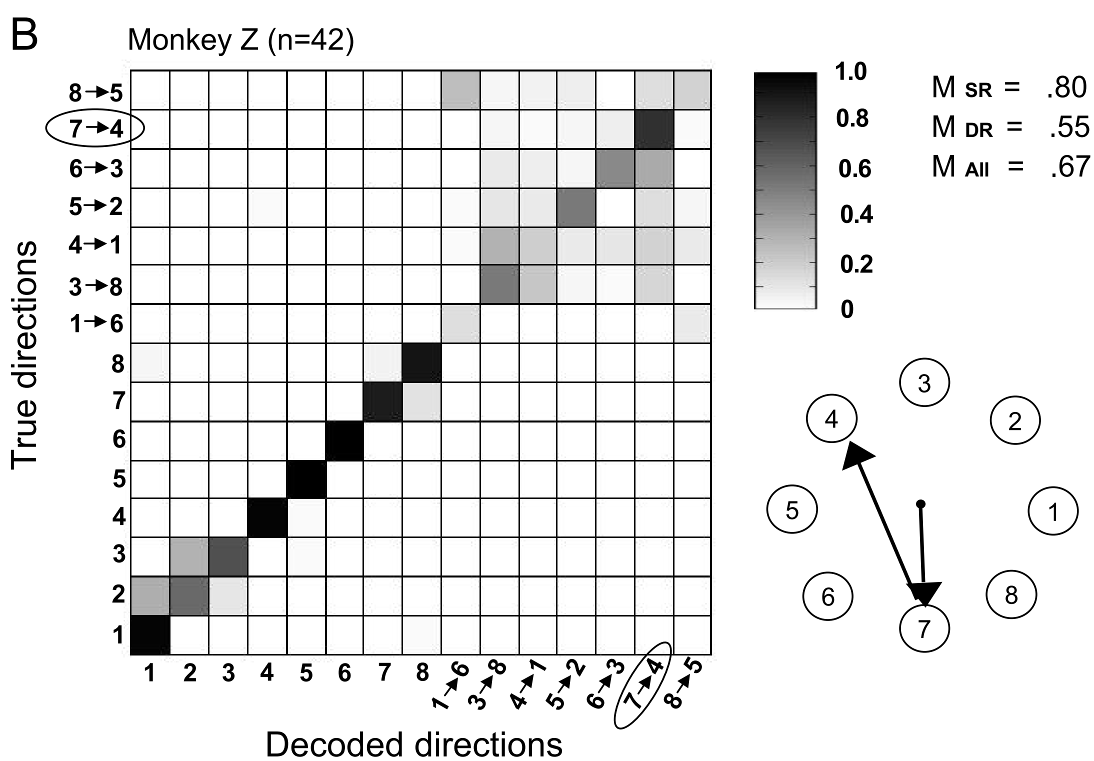
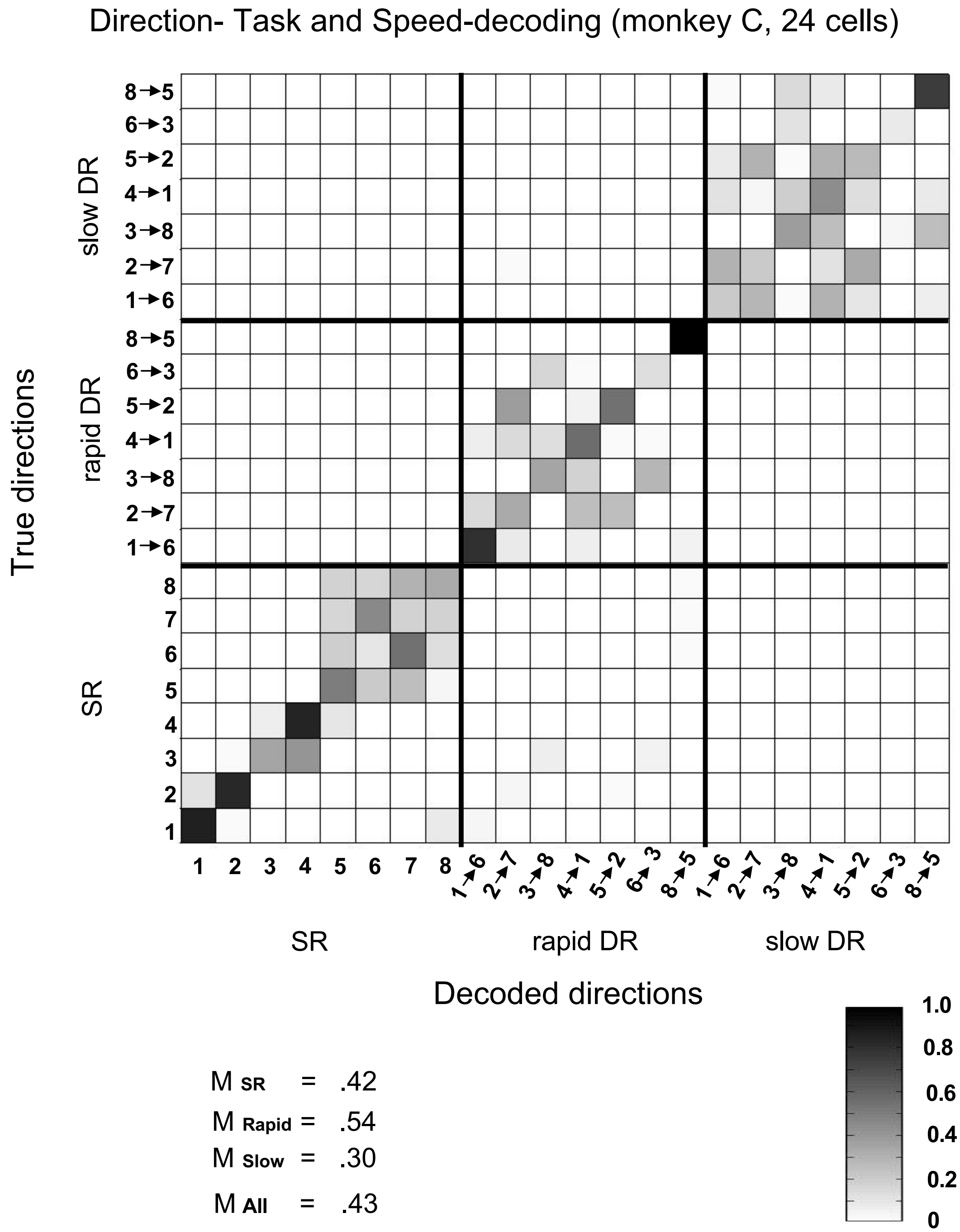
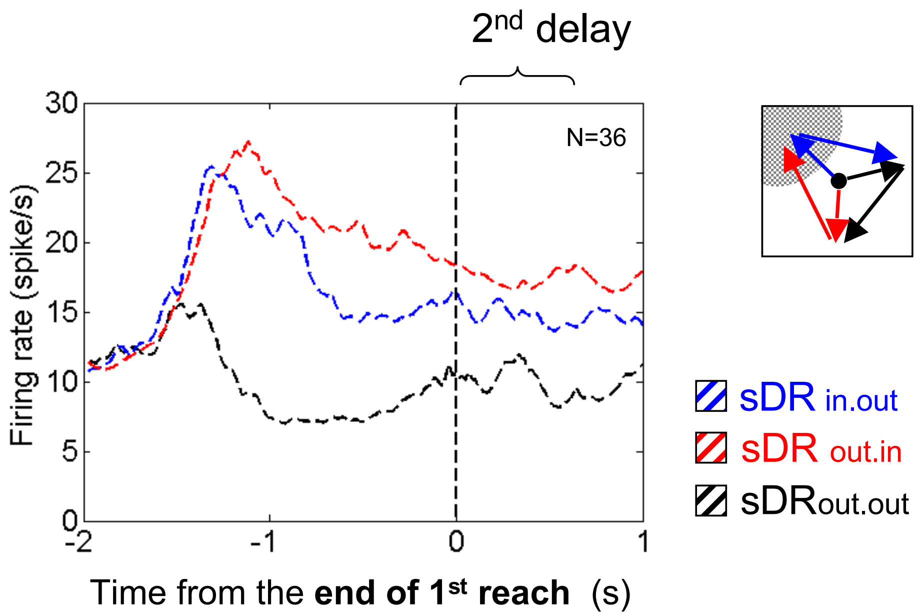

B
Monkey Z (n=42)
True directions
Decoded directions
8→5
7→4
6→3
5→2
4→1
3→8
1→6
1.0
0.8
0.6
0.4
0.2
0
M SR = .80
M DR = .55
M All = .67

Figure S2



M SR = .42
M Rapid = .54
M Slow = .30
M All = .43

Figure S3



2nd delay
30
25
20
15
10
5
0
Firing rate (spike/s)
N=36
-2
-1
0
1
Time from the end of 1st reach (s)
sDR in.out
sDR out.in
sDRout.out