

# N-backer: An auditory $n$ -back task with automatic scoring of spoken responses

Andrew F. Monk · Dan Jackson · Dea Nielsen · Elizabeth Jefferies · Patrick Olivier

Published online: 18 March 2011  
© Psychonomic Society, Inc. 2011

**Abstract** The  $n$ -back task is commonly used to load working memory (WM) in dual-task and neuroimaging experiments. However, it typically involves visual presentation and buttonpress responses, making it unsuitable for combination with primary tasks that involve vision and action, such as sequential object use and other tasks of daily living. The N-backer software presented here will automatically present and score auditory–verbal  $n$ -back sequences utilising the standard speech synthesis and recognition facilities that come with Microsoft Windows. Data are presented from an experiment in which 12 student participants carried out three tasks from the Naturalistic Action Test (NAT) while their attention was divided between the primary task and a continuous auditory–verbal 2-back secondary task. The participants' 2-back performance was scored in two ways: by hand, from video recordings, and automatically, using the software, allowing us to evaluate the accuracy of N-backer. There was an extremely high correlation between these scores (.933). The videos were also used to obtain a comprehensive error score for the NAT. Significantly more errors were made in the more complex NAT tasks when participants were 2-backing, as compared with when they were not, showing that the auditory–verbal  $n$ -back task can be used to disrupt sequential object use. This dual-task method may simulate the attentional deficits of patients with

brain injury, providing insights into the difficulties they face in tasks of daily living.

**Keywords** Executive working memory · Dual task experiments · Attentional deficits

## Introduction: The $n$ -back working memory task

The  $n$ -back task is widely used as a way of placing a continuous demand on working memory (WM) in neuroimaging and behavioural dual-task experiments (for reviews, see Jaeggi, Buschkuhl, Perrig, & Meier, 2010; Kane, Conway, Miura, & Colflesh, 2007; Owen, McMillan, Laird, & Bullmore, 2005). This task requires online monitoring, updating of WM, and rule-governed as opposed to familiarity-based decisions, and therefore loads the executive component of WM (e.g., Owen et al., 2005). Participants monitor a series of visually presented digits or letters and decide whether each item repeats the one presented  $n$  back in the sequence. Participants may be asked to press a response button when the current stimulus is an  $n$ -back repeat or, alternatively, may respond to each stimulus in turn by pressing one of two keys for yes/no. A key advantage of this method is that the demands on executive WM can be easily adjusted by changing the size of  $n$  (i.e., by comparing 0, 1, 2, or 3 back). In addition, researchers often contrast  $n$ -back tasks involving different materials and modalities—for example, verbal information (i.e., visually presented digits or letters) and spatial locations (e.g., Nystrom et al., 2000; Smith, Jonides, & Koeppe, 1996).

Dual-task studies frequently use  $n$ -back tasks to examine the influence of divided attention on a primary task (e.g., Baddeley, Hitch, & Allen, 2009; McKinnon & Moscovitch,

---

A. F. Monk (✉) · D. Nielsen · E. Jefferies  
Department of Psychology, University of York,  
York YO10 5DD, UK  
e-mail: a.monk@psych.york.ac.uk

D. Jackson · P. Olivier  
Digital Interaction Group, Culture Lab, Newcastle University,  
Newcastle upon Tyne NE1 7RU, UK

2007). Researchers have also compared verbal and spatial *n*-back tasks to determine the involvement of the verbal and visuospatial components of WM in different tasks or contexts (e.g., Baddeley et al., 2009). However, *n*-back tasks typically involve visual rather than auditory presentation (even for verbal materials) and, to facilitate the data analysis, almost invariably involve buttonpress as opposed to spoken responses. This prevents the *n*-back paradigm from being used in combination with primary tasks that involve vision and action, such as sequential object use—a difficulty addressed by the N-backer program presented here.

Sequential object use is thought to be executively demanding (Duncan, Emslie, Williams, Johnson, & Freer, 1996; Giovannetti, Schwartz, & Buxbaum, 2007; Morady & Humphreys, 2009) and is impaired in clinical populations with attentional deficits following stroke or dementia (Corbett, Jefferies, & Lambon Ralph, 2009; Giovannetti et al., 2008; Giovannetti, Schmidt, Sestito, Libon, & Gallo, 2006; Schwartz et al., 1998). Consequently, object use tasks are vulnerable to disruption by demanding secondary tasks (see Giovannetti et al., 2007; Morady & Humphreys, 2009). Dual-task studies provide a means of (a) exploring which aspects of sequential object use place the heaviest demands on attention and (b) evaluating the efficacy of technologies designed to assist everyday activities involving complex object use, such as cooking. We have therefore developed N-backer: an automated version of *n*-back, involving auditory presentation and *spoken responses*, allowing us to explore the impact of divided attention on sequential object use tasks that last for several minutes. Participants listen to a series of digits and attempt to repeat them with a lag of 1, 2, or 3 back. For example, on hearing the sequence 5, 2, 8, 9, 1 . . . , a participant performing 2-back would stay silent for the first two digits, then say “5” after hearing 8, “2” after hearing 9, and so on. This version of *n*-back, requiring spoken recall of each digit, has a number of advantages: It places a substantial and continuous load on attention and working memory, it overcomes the need to use a two-alternative forced choice decision (i.e., buttonpress responses to indicate whether the current digit matches the digit *n* back), and it allows for free movement of the hands and body. The computer is trained to recognise

participants’ spoken responses, and N-backer determines whether each spoken digit is correct or incorrect, avoiding the need for time-consuming manual coding. We examined the impact of this automated 2-back task on the Naturalistic Action Test (NAT; Schwartz, Buxbaum, Ferraro, Veramonti, & Segal, 2002).

## How N-backer works

### Principles of operation

N-backer utilises speech recognisers compliant with Microsoft’s Speech Application Programming Interface (SAPI), which is bundled with Microsoft Windows XP, Vista, and Windows 7. The participants wear a wireless headset with headphones and a boom microphone to facilitate free movement. They hear a sequence of synthesised digits through the headphones. This sequence may be of fixed length and read from a file, or randomly generated and continuous until terminated by the experimenter. The rate of presentation is set by the experimenter. The participant’s microphone picks up what he or she says, and speech recognition software records each digit spoken with a time stamp. Time boundaries are set to determine whether the digit spoken occurred within the parameters of the *n*-back task being performed. A free copy of the N-backer application can be downloaded from <http://irgen.ncl.ac.uk/go/nback/>.

A sample of the output from the program for a 2-back task is presented in Table 1. By following this table through time from the earliest presented digits (at the top of the table) to the later digits (at the bottom), it is possible to see how N-backer operates. In the first row, the computer presented the digit 5 (“spoken” by the system at speak time 7417). The system registered “5” as a potential response by the participant to this stimulus (at received time 11995). If no response is detected (see Section [Speech output and recognition](#)), “Received” is given as –1. *N* back is the difference between speak time and receive time, divided by the interstimulus interval—here, 1,500 ms. Using criteria detailed in Section [Automated scoring of n-back performance](#), automatic scoring algorithms judged this

**Table 1** Output from N-backer

	Speak time	Spoken	Receive time	Received	<i>n</i> back	Match	In time	Pass
	7417	5	11995	5	3.05	TRUE	TRUE	TRUE
	8917	7	13635	8	3.15	FALSE	TRUE	FALSE
	10417	6	15338	6	3.28	TRUE	TRUE	TRUE
“Speak time” and “Spoken” refer to the system “speaking” a digit. “Receive time” and “Received” refer to the software’s detection of a response from the participant	11917	1	16588	1	3.11	TRUE	TRUE	TRUE
	13417	2	0	–1	0	FALSE	FALSE	FALSE
	14917	5	18995	5	2.72	TRUE	TRUE	TRUE
	16417	7	20198	7	2.52	TRUE	FALSE	FALSE

response to be both correct as a 2-back response (Match = TRUE) and within the right time frame (In Time = TRUE; i.e., not 1 back or 3 back). Consequently, this response was considered to be correct (Pass = TRUE).

The information in Table 1 is provided by the N-backer software at the end of the experiment as a .CSV file, allowing response accuracy (i.e., the proportion of rows where Pass = TRUE) to be computed. Of course, there is a degree of uncertainty in any system involving the automatic recognition of speech, so in Section [An experiment using N-backer](#) we present an empirical case for the validity of the scores obtained.

In terms of a suggested procedure for using N-backer in a continuous task, pilot experiments showed that better recognition accuracy could be obtained by using the standard Microsoft tools to adjust the software to an individual participant's voice. These can be found in the Speech Control panel. The participant reads out continuous prose that turns grey as the program recognises it. A user voice profile can then be saved for that participant. It was also found advantageous to practice the participant on the 2-back task, to criterion performance, before the experiment started (see Section [Procedure](#) below).

#### Basic controls

Figure 1 shows the main interface window of the software. The toolbar controls should be largely self explanatory. The first group are standard Windows controls for saving the comma-separated value (CSV) text files that contain the data presented in the main panel (Speak time, Spoken, Receive time, etc.). CSV text files are supported by many analysis software tools.

With long sequences of  $n$ -backing, such as those used in the experiment described above, one would normally have the program generate a pseudorandom sequence of numbers

for itself (by selecting the Generate button, as in Fig. 1). For other uses of N-backer, one might want to generate the number sequences elsewhere and read them into the application. This can be achieved by generating a CSV file with just column 2 specified and the other seven columns entered as blank. In this case, the Generate button should be unselected, otherwise N-backer will append new numbers to the end of the specified sequence.

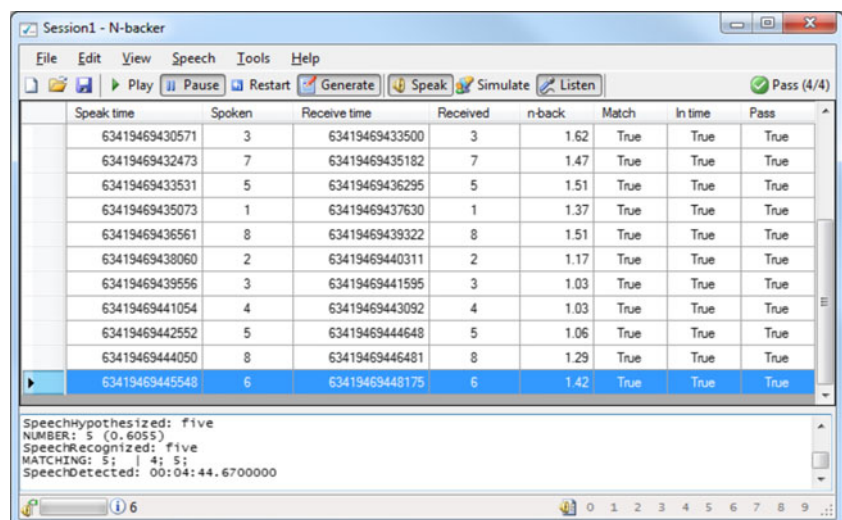
Play, Pause, Restart, Speak, and Listen are used to control presentation. The use of Simulate is explained in Section [Automated scoring of n-back performance](#). With the toolbar settings depicted in Fig. 1, pressing Play will cause the software to highlight the active row and speak the number in that row while recording a timestamp. A speech synthesiser is used to say the numbers aloud to the participant. The active row will advance automatically at the interval defined by the operator. The  $n$ -back interstimulus interval is the first setting in the control panel, here 1,500 ms (see Fig. 2).

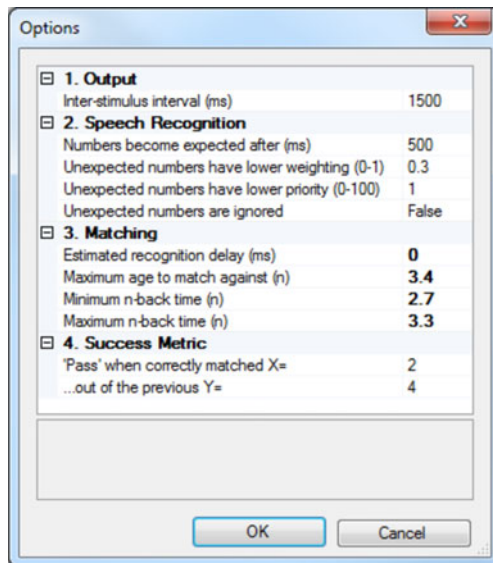
#### Speech output and recognition

Speech recognition software is used to recognise the participant's spoken numbers. A fixed set of single-token "recognition grammars" are used, one for each of the numerals 1 to 9. The speech recogniser is configured to only perform recognition against these grammars, preventing it from trying to match against any other possible speech.

We have found that under normal circumstances, if the standard Microsoft tools to adjust the software to an individual participant's voice are used, simply limiting the speech recognised to numerals in this way is sufficient to get good hit rates and low false negatives. However, we have provided settings to cater for other circumstances, where it may be necessary to increase the recognition rate

**Fig. 1** N-backer main interface window





**Fig. 2** N-backer settings window

of correct answers, at the expense of higher false positive rates (see Fig. 2 for these control panel settings). These use the fact that some numbers are expected and some unexpected. “Unexpected numbers” are numbers that are not expected to be spoken by the participant, since they would not be a correct response given the numbers earlier in the sequence. Numbers become “expected” after a specified amount of time after speak time (the “Numbers become expected after (ms)” setting in Fig. 2), and stay expected until that prompt is matched with a correct answer, or until the presented number exceeds the maximum matching age (“Maximum age to match against (n)” setting).

Unexpected numbers can have lowered recognition weighting (“Unexpected numbers have lower weighting (0–1)” in Fig. 2, set to less than 1), making them less likely to be matched than expected numbers, or lower recognition priority, so that unexpected numbers will lose in a recognition tie (“Unexpected numbers have lower priority (0–100),” set to less than 100). In very extreme circumstances, one might need to set the recogniser to

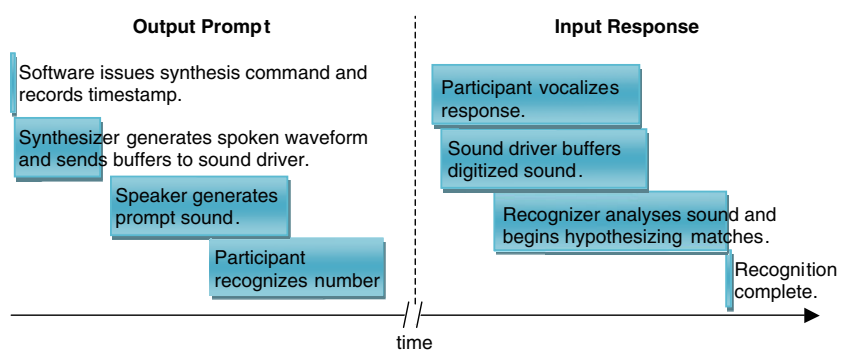
ignore unexpected numbers entirely (“Unexpected numbers are ignored,” set to True). This will result in the Received column registering –1 (unknown) rather than a best-matched digit in the case of an inappropriate response.

#### Automated scoring of n-back performance

Recognised numbers are compared against any outstanding unmatched numbers, up to the configured maximum matching age (“Maximum age to match against (n)” setting in Fig. 2). Incorrect (recognised but unmatched to an expected number) responses are stored alongside their probable target (i.e., the *n*-back number that should have been produced). A timestamp for when the spoken response was recognised is recorded with the number spoken. This allows for the computation of the delay between the timestamp recorded when the software issued a command to synthesise the number (“Speak time” in Fig. 1) and the participant’s response. This difference is expressed as *n* in the “n-back” column—that is, the difference in milliseconds is divided by the interstimulus interval, here 1,500 ms.

For example, consider the first row of data in Table 1. The routine to speak “5” was called at time 7417. Then, 4,578 ms later, at 11995 (Table 1, column 3), the program registered that “5” (column 4) had been spoken, so that “n back” is recorded as 3.05 (column 5). This would appear to be too late, and should thus have been scored as In time = FALSE; in fact, it is scored as In time = TRUE because the limits for *n* back to be judged as in time were set to being between 2.7 and 3.4. This rather confusing situation arises because the first version of the software used in the experiment described above had no other way to take account of the delay in the system caused by the time taken by the participant to say the digit and for the speech recogniser to recognise it. These delays are schematically depicted in Fig. 3. The recognition delay is the time between the points at which the participant begins to vocalise the number and at which recognition is complete. This depends on processor speed and the time it takes to say the number. By setting the “Minimum n-back time”

**Fig. 3** Schematic timing diagram illustrating delay overhead between prompt and response (not based on data)





parameter to 2.7 (4,100 ms), and “Maximum  $n$ -back time” to 3.3 (5,100 ms), we are effectively adding around 1,000 ms for the recognition delay.

With the initial version of the software, setting the limits for  $n$  back to be judged as “in time” required some trial and error. The latest version of the software includes an extra setting with which the user can specify the recognition delay. To estimate this delay, this new version of N-backer also has a Simulate mode (see the toolbar at the top of main interface window depicted in Fig. 1). Selecting this mode simulates the responses of a highly accurate 1-back participant by speaking the prompt very quietly and the response loudly at the correct time—the microphone is adjusted to hear the response, and any deviation from 1 back can be adjusted for. The recognition delay (“Estimated recognition delay (ms)” setting in Fig. 2) should only have to be set once for a particular computer configuration.

Finally, to provide continuous feedback to an experimenter on the participant’s performance throughout a study, a simple “success metric” was implemented. The rightmost item in the toolbar at the top of the window depicted in Fig. 1 signals pass or fail in terms of “ $x$  correct matches out of the previous  $y$ ”—for example, where  $x = 2$  and  $y = 4$ , the user will “pass” if two or more of the last four matches are within the allowed  $n$ -back limits.

The application is written in the C# programming language on Microsoft .NET Framework 3.5 and uses SAPI (minimum version 5.1) speech synthesis and recognition, found on Windows XP and later platforms.

### An experiment using N-backer

The purpose of this experiment was to assess the accuracy of automated  $n$ -back scoring in a continuous dual-task procedure using this software. The primary tasks for the participants to carry out while  $n$ -backing were provided by the NAT (Schwartz et al., 2002). The NAT is widely used to assess patients with brain injury and includes three subtasks of varying difficulty. Subtask 1 requires two highly familiar tasks, making toast and coffee, to be performed simultaneously. Subtasks 2 and 3 are harder, since their goals (wrapping a present and packing a lunchbox and schoolbag) are slightly less familiar. Moreover, these tasks are performed in the presence of irrelevant distractor objects that must be ignored. Subtask 3 is arguably the hardest of all, because it involves two overlapping tasks, packing a lunch box *and* a school bag, and requires participants to avoid packing irrelevant objects. The first two subtasks took between 2 and 4 min to complete, and the last between 3 and 5 min.

These relatively long primary tasks were selected as a hard test for the software and as an example of an

experimental procedure in which automated scoring would be most valuable. In particular, as activities of daily living, they may be judged to have greater ecological validity than simpler but more artificial tasks. We were also interested in the effect of  $n$ -backing on performance of the NAT, which has a standardised scoring scheme.

## Method

### Design

Each participant completed one subtask under normal (control) conditions and the other two while  $n$ -backing, in a balanced design. This yielded  $n$ -back runs for 8 participants for each of the three subtasks and normal runs for 4 participants for each subtask. This compromise maximised the data available for assessing the accuracy of automated scoring while still permitting a meaningful comparison of  $n$ -back and control trials.

### Participants

A total of 12 participants were recruited through personal contacts. They were all graduate or undergraduate students at the University of York between the ages of 18 and 25. Eight were female.

### Procedure

Table 2 gives the basic structure of the experiment. After completing a consent form, participants trained the speech recognition software to recognise their voice (see above). They were given instructions explaining how to 2-back (see the Appendix), and then they listened to a synthesised digit sequence to familiarise them with its sound and pace. This was followed by practice at 2-backing. Participants continued to practice the 2-back task, with breaks if necessary, until they could produce at least 8 consecutive

**Table 2** The basic structure of the experiment

1.	Train speech recognition
2.	Instructions and demonstration of N-backer digit sequence
3.	Practice 2-back to criterion
4.	Practice 2-back while sorting playing cards
5.	First baseline: 2-back task on its own for 3 min
6.	First NAT subtask
7.	Second NAT subtask
8.	Third NAT subtask
9.	Second baseline: 2-back task on its own for 3 min

Two NAT subtasks were carried out while 2-backing, and one was performed under normal conditions (see the Design section)

responses without error and 20 responses with 80% accuracy. Furthermore, they needed to demonstrate that they were able to get quickly back into the sequence (within 3–4 digits) if they paused or made an error—for example, switching to 1-backing or 0-backing. Feedback was given about accuracy and the types of errors made, particularly if the participant slipped into 0- or 1-backing. When practiced to the criteria described above, participants were asked to 2-back while sorting a pack of playing cards into suits, as an introduction to the dual-task method, and then the first baseline 2-back measurement was made. This involved participants 2-backing in the absence of a primary task until the experimenter stopped them after 3 min.

Before performing each NAT subtask, participants left the room while the test objects for that subtask were set out in their standard positions on a U-shaped table. On participants' return, they were given the standard NAT instructions for the subtask. The NAT also has instructions about how to answer participants' questions. In general, these instructions are just to reiterate the overall task and to explain that “everything you need is out here.” The “time-based cues” for the NAT were not used (e.g., prompting after 1 min of unproductive behaviour). At the end of the task, the participant either indicated of their own accord that they had finished or they were asked, “Are you finished?” at which point the program was stopped and the output saved. Upon completing the third subtask, participants provided a second, 3-min 2-backing-only baseline, so that any improvements in 2-back performance over the course of the experiment could be measured.

### Apparatus

The N-backer software was run on a Dell PC running Windows XP Version 2002, Service Pack 3. This had an Intel Pentium dual CPU at 2.2 GHz and 3 GB of RAM. The stimuli were presented and responses registered via a LogitechClearChat wireless USB headset with over-ear phones and a boom microphone.

In normal use, such a headset would be the only necessary additional hardware required. For the purpose of this experiment, which was to assess the accuracy of the N-backer software, sessions were video recorded, and participants wore two extra microphones, the input from which were fed into the separate audio channels on the video camera. A clip microphone on the participant's shirt recorded verbal responses, and a small microphone in the left headphone recorded the stimuli output by the N-backer software. There was some crosstalk on this channel because of bone conduction, but it was still possible to clearly identify the digits generated. The objects for the NAT were as specified in the test manual.

**Table 3** N-backing accuracy as scored by N-backer (AUTO) and by hand from the audio recording

	B1	NAT1	NAT2	NAT3	B2
AUTO					
Mean	83%	72%	70%	68%	86%
Std dev	13%	11%	17%	10%	12%
Minimum	53%	58%	45%	50%	59%
HAND					
Mean	85%	74%	68%	70%	87%
Std dev	14%	9%	13%	11%	12%
Minimum	52%	59%	47%	49%	54%
<i>r</i>	.95	.96	.86	.92	.95
<i>N</i>	12	8	8	8	12

B1 and B2 are the initial and final baseline measurements, and NAT1–NAT3 refer to the NAT subtasks. *r* is the correlation between AUTO and HAND

The N-backer settings used in the experiment (Fig. 2; see Section [How N-backer works](#) in general for explanations) were Inter-stimulus interval (ms) = 1500; Numbers become expected after (ms) = 500; Unexpected numbers have lower weighting = 0.3; Unexpected numbers have lower priority = 1; Unexpected numbers are ignored = False; Maximum age to match against (n) = 3.4; Minimum n-back time (n) = 2.7; and Maximum n-back time (n) = 3.3. Note that the latter two parameters are set deliberately high, because there was no estimated recognition delay in this version of the software (see Section [Automated scoring of n-back performance](#) for an explanation).

### Results

#### N-backer reliability and accuracy

In order to assess the accuracy with which N-backer scored the participants' 2-backing performance, the videos had to be scored by hand (this, of course, would not be necessary in the normal use of N-backer). Manual scoring was done by the third author, who had considerable experience of

**Table 4** NAT summary scores (out of 6) for each subtask for the 4 participants under control conditions and the 8 participants under 2-back conditions

		Mean	Std Dev	Minimum
Subtask 1	Control	6.00	0	6
	2-back	5.50	0.93	4
Subtask 2	Control	6.00	0	6
	2-back	5.25	1.49	2
Subtask 3	Control	3.50	1.00	2
	2-back	4.25	1.28	2

**Table 5** Comprehensive error scores for each subtask for the 4 participants under control conditions and the 8 participants under 2-back conditions

Mean = Mean number of errors produced per subtask per participant. M–W  $U$  = Mann–Whitney  $U$  test

		Mean	Std Dev	Mean Rank	M–W $U$	$Z$	$p$
Subtask 1	Control	0.50	0.58	7.25	13	0.612	.54
	2-back	0.38	0.74	6.13			
Subtask 2	Control	0.00	0.00	3.50	4	2.189	.029
	2-back	1.50	1.20	8.00			
Subtask 3	Control	1.50	0.58	3.50	4	2.109	.035
	2-back	2.88	0.99	8.00			

doing this with shorter sequences on another project. Scoring 2-back recordings without the help of a computer is hard, since one can make only approximate judgements of timing. It is quite easy, however, to transcribe the sequence of digits spoken by the participant. Manual scoring was thus achieved by matching this transcript to the digits generated by N-backer (Table 1, column 2).

Discrepancies between the transcription and the digits generated by N-backer commonly occurred after participants had slipped into 1- or 0-back responding (i.e., when they repeated the digit they had just heard). Participants would then pause for 2 digits and then start 2-backing again. This pause caused all of the subsequent digits in the transcript to be two items out of place, so these needed to be shifted down in the spreadsheet. Similar adjustments to the transcripts were made wherever necessary to match the transcription to the digits generated by N-backer.

Having achieved a manual transcript in this way, a by-hand score could be computed by counting the proportion of transcribed digits that were correctly 2 back. These percentages are presented in Table 3, along with the automatic accuracy score generated by N-backer. The means and minimum scores across participants are very similar for the two types of scoring, and there is no suggestion that either is more or less lenient. This similarity is reflected in the correlations between the proportions of transcribed digits that were correctly 2 back, computed for both manual and automatic scoring. These are reported in Table 3 and are very high. If the four measurements obtained from each participant are treated as if they were independent, an overall  $r$  of .93 ( $N = 48$ ) is obtained.

The Pearson correlations reported above characterise the interrater reliability of the score (proportion of transcribed digits that were correctly 2 back) rather than interrater agreement across individual judgements, as measured by

Cohen's kappa. Interrater reliability will be of most interest to investigators using N-backer to obtain an  $n$ -backing score. Interrater agreement is of interest to investigators who wish to use N-backer to look at detailed  $n$ -backing behaviour—for example, exactly where in the task  $n$ -backing breaks down. Kappa was computed for each of the 24  $n$ -back records—that is, for the two  $n$ -back conditions experienced by each of the 12 participants. The mean kappa was .72 ( $s = .17$ ).

#### NAT errors

The NAT is scored by counting the number of task steps accomplished. This accomplishment score is combined with an error score to give a summary score. Few of our participants produced major errors or step omissions, so these scores were close to ceiling for Subtasks 1 and 2 (see Table 4). Subtask 3 (lunchbox and schoolbag) included some steps that were less intuitive to our participants, and here the NAT scores were lower. However, there was little difference in the mean scores between the 2-back and control (no 2-back) conditions. It would appear that 2-backing had little effect on performance on this clinically diagnostic scale. The NAT manual reports mean scores (summing across all three subtasks) of 17.3 ( $SD = 1.2$ ) for control subjects. All of the scores reported in Table 4 are close to this range.

In addition to this clinically diagnostic NAT score, the test also provides a scoring scheme for a more detailed analysis, the comprehensive error score. Comprehensive error scores were calculated separately for each subtask using the video recordings. Table 5 gives the mean comprehensive error score for each subtask for participants performing under control conditions (without concurrent 2-back task) and under 2-back conditions. Because of the counterbalanced design adopted, there were 8 participants in the 2-back group and 4 participants in the control group. The participants in

**Table 6** Combined comprehensive error scores plus toying and hesitations for the 4 participants under control conditions and the 8 participants under 2-back conditions

Mean = mean number of errors produced per subtask per participant. M–W  $U$  = Mann–Whitney  $U$  test

		Mean	Std Dev	Mean Rank	M–W $U$	$Z$	$p$
Subtask 1	Control	0.50	0.58	3.50	4	2.082	.037
	2-back	3.50	2.39	8.00			
Subtask 2	Control	0.00	0.00	2.50	0	2.787	.005
	2-back	5.25	3.20	8.50			
Subtask 3	Control	1.75	0.50	2.5	0	2.787	.006
	2-back	7.00	1.69	8.5			

each group varied across the subtasks. For this reason, and given the small sample sizes involved, Mann–Whitney  $U$  tests were carried out for each subtask. This analysis found that 2-backing had little effect on the first subtask (toast and coffee), which was arguably the easiest, but produced more errors in the two other subtasks (wrapping a present and preparing a lunchbox and schoolbag; see Table 5).

The authors of the NAT explicitly exclude “prolonged struggles to manipulate and use items and misreaching that falls short of taking the object” from the error scores. This misses “toying,” a commonly observed behaviour of people with executive impairment following stroke or dementia (e.g., Corbett et al., 2009; Wherton & Monk, 2010), where patients repeatedly touch, move, or aimlessly pick up and then replace an object. This type of directionless and inefficient behaviour was observed when participants were performing 2-back concurrently with the NAT subtasks. Therefore, to get a more sensitive measure of the disruption to behaviour caused by 2-backing, a second error score was computed that included toying. *Toying* was defined as any action in which an object was picked up and put back down without being used. Actions judged as intended to clear space on the work surface or clean up after a task (e.g., closing the butter or putting the lid back on the juice bottle) were not categorised as toying. In addition, hesitations—defined as when the participant stopped both 2-backing and the primary task for more than 2 s—were coded. Table 6 presents means for total numbers of errors (i.e., the comprehensive error score plus toying and hesitations). All three subtasks show significant effects of 2-backing on this score.

## Discussion

This article presents a methodology for auditory–verbal  $n$ -back tasks with automated scoring, using software we have developed called N-backer. The empirical findings demonstrate that the N-backer program produces reliable results, even when N-backer is used as a secondary task and combined with primary tasks lasting several minutes in duration—circumstances in which manual coding is time consuming and challenging. Automated scoring was highly correlated with the results obtained by manual coding of participants’ spoken responses ( $r = .933$ ).

A 2-back version of this task was used with young healthy adults carrying out the NAT, a standardised neuropsychological assessment used extensively with brain-injured patients (Schwartz et al., 2002). While the samples are very small, this task was sufficiently demanding to significantly disrupt sequential object use. The greatest disruption was seen on the more executively demanding subtasks, which involved multitasking and required that distracting objects be ignored. Low-level errors, hesitations, and toying behaviour—in which

participants aimlessly picked up and put down objects—occurred more frequently under dual-task conditions.

Patients with executive or attentional deficits following brain injury also perform poorly on the same tasks within the NAT, and also show frequent hesitations and toying (Corbett et al., 2009). However, complete action additions and omissions were relatively rare in our healthy participants, and hence there were no striking effects of 2-backing on the diagnostic “NAT score.” This null result is difficult to interpret, given the very low  $N$  in this comparison, but it is clear that our procedure disrupts sequential object use without mimicking all of the symptoms of executive working memory and attention deficits observed in brain-injured patients.

The results reported above suggest that N-backer is a useful tool for exploring the role of executive control in tasks involving vision and/or action, particularly sequential object use tasks that place significant demands on executive mechanisms. We have shown that N-backer produces accurate results over long test durations for healthy young individuals, who are broadly representative of the participants that are typically recruited in  $n$ -back studies. Nevertheless, other potential uses for the software might include investigations of sequential object use in older adults (with whom our methods could be used to explore the benefits of technological and ergonomic aids designed to support the completion of complex everyday tasks) and in settings outside the laboratory (in tasks such as cooking and personal care in participants’ homes). In pursuing these research questions, it will be important to establish the degree to which N-backer can cope with suboptimal speech, recall that includes long pauses, and extraneous background noises.

**Author Note** Funding for the software development was provided in part from the VERITAS project, awarded under the EU 7th Framework programme. We thank Azizah Almaghyuli at the University of York for her help in running and analysing the experiment, and Faye Corbett for her advice about the analysis.

## Appendix: 2-backing instructions

You will hear a continuous series of random digits from 1–9.

Listen to the first two numbers without responding.

Then, for each number you hear, say the digit you heard 2 digits ago.

For example:

“9” –> listen

“6” –> listen

“1” –> Say 9

“8” –> Say 6

“2” –> Say 1

“1” –> Say 8



If you get lost, a good strategy to get back into the sequence is to stop for a moment, collect yourself, focus on a number you hear and then restart. Try to get back into the sequence as fast as possible.

It is critical that you make as few errors as possible on the number task. This may mean that you make a few errors on the action sequence task; however this is to be expected.

Please ask the researcher if there is anything you are unsure about.

## References

- Baddeley, A. D., Hitch, G. J., & Allen, R. J. (2009). Working memory and binding in sentence recall. *Journal of Memory and Language*, *61*, 438–456.
- Corbett, F., Jefferies, E., & Lambon Ralph, M. A. (2009). Exploring multimodal semantic control impairments in semantic aphasia: Evidence from naturalistic object use. *Neuropsychologia*, *47*, 2721–2731.
- Duncan, J., Emslie, H., Williams, P., Johnson, R., & Freer, C. (1996). Intelligence and the frontal lobe: The organisation of goal-directed behaviour. *Cognitive Psychology*, *30*, 257–303.
- Giovannetti, T., Bettcher, B. M., Brennan, L., Libon, D. J., Kessler, R. K., & Duey, K. (2008). Coffee with jelly or unbuttered toast: Omissions and commissions are dissociable aspects of everyday action impairment in Alzheimer's disease. *Neuropsychology*, *22*, 235–245.
- Giovannetti, T., Schmidt, K., Sestito, N., Libon, D., & Gallo, J. (2006). Everyday action in dementia: Evidence for differential deficits in Alzheimer's disease versus subcortical vascular dementia. *Journal of the International Neuropsychological Society*, *12*, 45–53.
- Giovannetti, T., Schwartz, M. F., & Buxbaum, L. J. (2007). The coffee challenge: A new method for the study of everyday action errors. *Journal of Clinical and Experimental Neuropsychology*, *29*, 609–705.
- Jaeggi, S. M., Buschkuhl, M., Perrig, W. J., & Meier, B. (2010). The concurrent validity of the *N*-back task as a working memory measure. *Memory*, *18*, 394–412.
- Kane, M. J., Conway, A. R., Miura, T. K., & Colflesh, G. J. (2007). Working memory, attentional control, and the *N*-back task: A question of construct validity. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *33*, 615–622.
- McKinnon, M. C., & Moscovitch, M. (2007). Domain-general contributions to social reasoning: Theory of mind and deontic reasoning reexplored. *Cognition*, *102*, 179–218.
- Morady, K., & Humphreys, G. W. (2009). Comparing action disorganization syndrome and dual-task load on normal performance in everyday action tasks. *Neurocase*, *15*, 1–12.
- Nystrom, L. E., Braver, T. S., Sabb, F. W., Delgado, M. R., Noll, D. C., & Cohen, J. D. (2000). Working memory for letters, shapes and locations: fMRI evidence against stimulus-based regional organisation in human prefrontal cortex. *Neuroimage*, *11*, 424–446.
- Owen, A. M., McMillan, K. M., Laird, A. R., & Bullmore, E. (2005). *N*-back working memory paradigm: A meta-analysis of normative functional neuroimaging studies. *Human Brain Mapping*, *25*, 46–59.
- Schwartz, M. F., Buxbaum, L. J., Ferraro, M., Veramonti, T., & Segal, M. (2002). *NAT: Naturalistic Action Test*. Bury St. Edmunds: Thames Valley Test Co.
- Schwartz, M. F., Montgomery, M. W., Buxbaum, L. J., Lee, S. S., Carew, T. G., Coslett, H. B., et al. (1998). Naturalistic action impairment in closed head injury. *Neuropsychology*, *12*, 13–28.
- Smith, E. E., Jonides, J., & Koeppel, R. A. (1996). Dissociating verbal and spatial working memory using PET. *Cerebral Cortex*, *6*, 11–20.
- Wherton, J. P., & Monk, A. F. (2010). Problems people with dementia have with kitchen tasks: The challenge for pervasive computing. *Interacting with Computers*, *22*, 253–266.