tvst

Article

A Method for Reducing the Number of Presentations in Perimetric Test Procedures

Andrew Turpin[1] **and Allison M. McKendrick**[2]

[1] School of Computing and Information Systems, The University of Melbourne, Melbourne, Australia
[2] Department of Optometry and Vision Sciences, The University of Melbourne, Melbourne, Australia

Correspondence: Andrew Turpin, School of Computing and Information Systems, The University of Melbourne, Melbourne, Victoria 3010, Australia.
e-mail: aturpin@unimelb.edu.au

Received: December 4, 2021
Accepted: March 12, 2022
Published: April 15, 2022

Keywords: perimetry; algorithms; testing procedures; visual fields

Citation: Turpin A, McKendrick AM. A method for reducing the number of presentations in perimetric test procedures. Transl Vis Sci Technol. 2022;11(4):14, https://doi.org/10.1167/tvst.11.4.14

Purpose: To introduce a new method (ARBON) for decreasing the test time of psychophysical procedures and examine its application to perimetry.

Methods: ARBON runs in parallel with an existing psychophysical procedure injecting occasional responses of seen or unseen into that procedure. Using computer simulation to mimic human responses during perimetry, we assess the performance of ARBON relative to an underlying test procedure and a version of that procedure truncated to be faster. Simulations used 610 normal eyes (age 20 to 80 years) and 163 glaucoma eyes (median mean deviation = −1.81 dB, 5th percentile = +2.14 dB, 95th percentile = −22.55 dB). Outcome measures were number of presentations and mean absolute error in threshold estimation. We also examined the probability distribution of measured thresholds.

Results: ARBON and the Truncated procedure reduced presentations by 16% and 18%, respectively. Mean error was increased by 8% to 10% for the Truncated procedure but decreased by 5% to 7% for ARBON. The probability distributions of measured thresholds using ARBON overlapped with the Underlying procedure by over 80%, whereas the Truncated procedure overlapped by 50%.

Conclusions: ARBON offers a principled method for reducing test time. ARBON can be added to any existing psychophysical procedure without requiring any change to the logic or parameters controlling the procedure, resulting in distributions of measured thresholds similar to those of the underlying procedure.

Translational Relevance: ARBON can be added to a perimetry test procedure to speed up the test while largely preserving the distribution of returned sensitivities, thus producing normative data similar to the data for the original, underlying perimetric test.

Introduction

Clinical visual field testing typically involves collecting light-increment thresholds at 50 or more locations across the central visual field (usually 24°–30°). Sensitivity is measured using one of a family of psychophysics algorithms, which in this context are referred to as perimetry algorithms. These include various forms of staircase procedures (for example, Full Threshold strategy,[1] Dynamic strategy,[2] and GATE[3]); Bayesian procedures (for example, variants of the ZEST procedure[4–6]); hybrids of the two (such as the SITA family of algorithms[7–10]); and other approaches (for example, PASS[11] and TOP[12]). All of these algorithms have governing parameters that determine things such as when the test terminates, what order the locations in the visual field are tested, and the luminance level of stimuli presented. The selection of these parameters controls the trade-off between precision and accuracy of threshold estimation and test time. Usually, altering the parameters to make testing faster will reduce the precision and accuracy of the threshold measurements. For example, a key difference between the SITA Standard test procedure and SITA Fast is an alteration to the termination parameter (ERF - error-related factor),[9] which results in decreased test time for SITA Fast but also some differences in the distributions of the returned threshold estimates.[13]

translational vision science & technology

In this paper, we explore a way to decrease test times of psychophysical testing algorithms without altering any of the parameters or logic of the algorithm itself. We introduce ARBON (Artificial Responses Based on Neighbors), which runs in parallel with any underlying procedure and occasionally injects artificial responses ("seen" or "unseen") at a location based on the status of the surrounding locations or neighbors. We experiment with this new approach in the context of perimetry by simulating a typical visual field test using the ZEST algorithm both with and without ARBON, showing that it reduces the number of presentations required without altering the underlying logic or parameters of the algorithm. A benefit of using such an approach is that, because the underlying algorithm is left intact, normative data collected using the faster approach should generally be closer to the data collected using the original than if parameters of logic are altered to produce a faster test.

Methods

Overview of Approach and Description of Algorithm

All Yes–No (seen–unseen) psychophysical algorithms can be expressed as a binary decision tree where each node in the tree is the level at which a stimulus is presented and two branches lead from a node to the next presentation level after either a Yes or No response. Because visual field procedures are short, the decision tree can often be visualized. Figure 1 shows such a decision tree for a ZEST procedure using parameters typical for those in perimetry.[14,15] In particular, the procedure stops when the standard deviation of its probability distribution of likely thresholds at a location drops below 2.0 dB, with a maximum of 10 presentations allowed. The exact details of the procedure generating the specific decision tree are not particularly important for this manuscript; this underlying algorithm serves to illustrate the ARBON method, which can be applied to any psychophysical algorithm. This particular tree would require further testing and engineering before it could be used in perimetry. We refer to it hereafter as the Underlying procedure.

As observed in the introduction, one way of achieving a faster test procedure is to terminate the procedure earlier or at a shallow depth in the tree. For example, the tree could be truncated at a fixed depth such as in the Humphrey Matrix Perimeter.[16] The red nodes in Figure 1 show the Underlying ZEST procedure if the stopping standard deviation is raised to 2.5 dB rather than 2.0 dB. As can be seen, paths in the tree are shorter, so the number of presentations to determining a threshold is smaller. For example, if the subject responds "seen" to 25 dB on the first presentation and "seen" to 29 dB on the second presentation, the shorter procedure stops and reports a threshold of 31 dB, whereas the original presents one more stimulus of 31 dB before returning either 30 or 32 dB. We refer to this shorter procedure as Truncated.

One problem with reducing the number of presentations by altering the algorithmic parameters is that it is possible that some threshold values that were previously obtainable using the procedure can no longer be reached. In the example shown, there are eight decibel values that could be returned as a final threshold by Underlying that are no longer available in Truncated (1, 7, 10, 13, 14, 19, 30, and 31 dB). This could result in a change in the distributions of normative data between the new and old procedures and might also change test-retest limits that are used to determine the probability of change in visual field data.

ARBON takes a different approach to reducing the number of stimulus presentations at any one location. We leave the underlying decision tree unaltered, but occasionally follow a Yes or No branch during the test based on the status of neighboring locations rather than as the result of a subject's response. That is, we infer an artificial response assuming that the current location will have a final threshold close to its neighbors.

To introduce the procedure precisely we define the *range of possible thresholds* (ROPT) of a node in the tree to be the range of all the possible final thresholds that could be reached from the node. Throughout, we use the standard Cartesian coordinate notation (x, y) to refer to nodes in the tree of Figure 1, and the notation $[a, b]$ to indicate a range of values that includes both a and b. As an example of ROPT, the node in Figure 1 at (8, 27) can lead to threshold values 26, 28, and 31, so the ROPT is [26, 31]. Similarly, the root node (0, 25) has a ROPT of [0, 32]. Further, we define the ROPT of a group of nodes as the range of their individual possible ranges. That is, the ROPT of a group of ROPTs $[a_1, b_1]$, $[a_2, b_2]$, …, $[a_n, b_n]$ is $[\min(a_1, a_2, \ldots, a_n), \max(b_1, b_2, \ldots, b_n)]$.

At any stage in the test, each location in the visual field that is not complete (i.e., we are not at a leaf node in the decision tree) has three possible ROPTs of interest to ARBON.

1. ROPT_Yes is the ROPT at the end of the Yes branch from the node representing the range of final thresholds that would be possible if a Yes response was recorded at this location.
2. ROPT_No is the ROPT at the end of the No branch from the node representing the range of

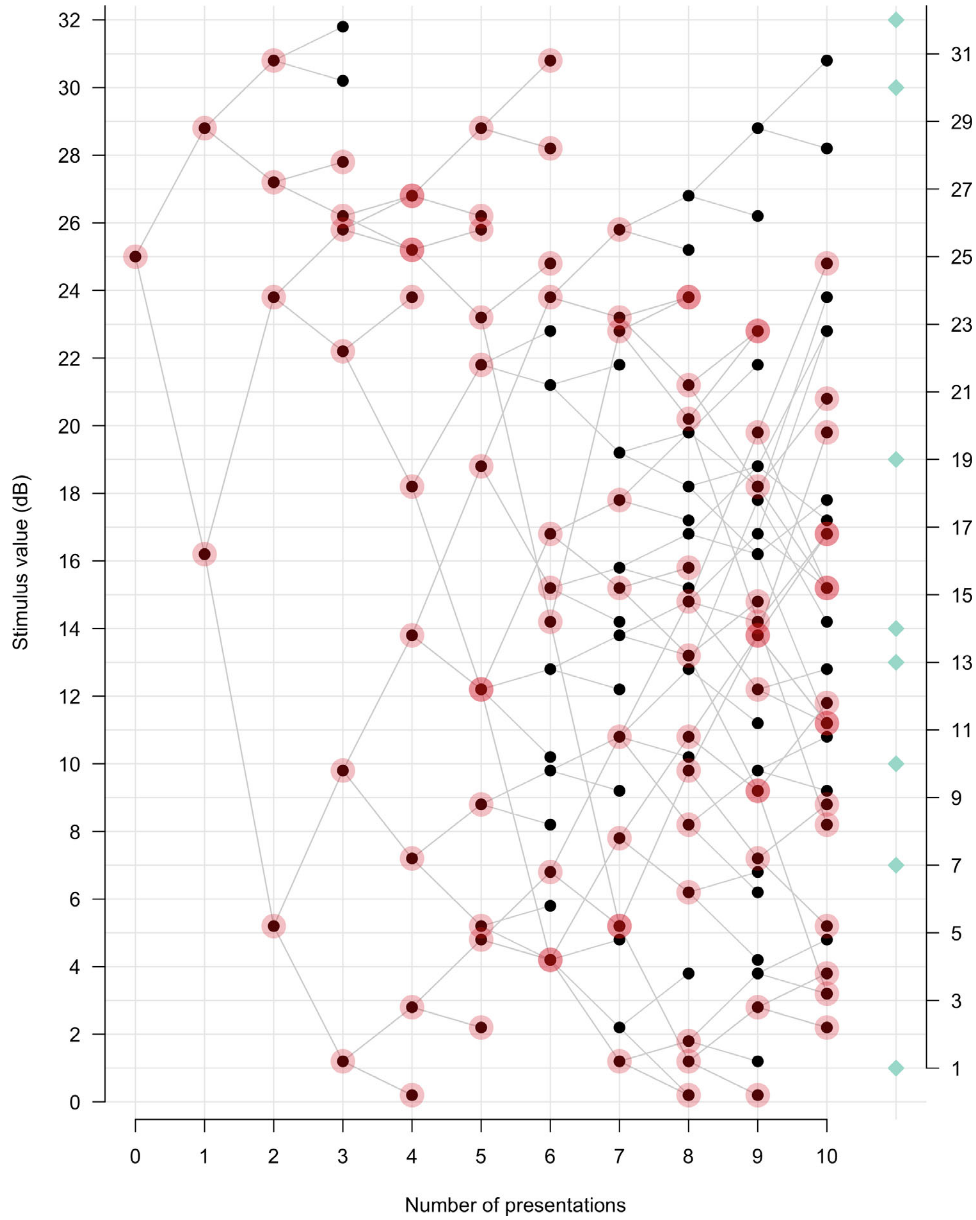

Figure 1. The decision tree used as an illustration of an underlying, short psychophysical procedure in this study. The *x*-axis shows the depth in the tree, or number of presentations, and the *y*-axis shows the stimulus value of each node represented as a *black dot* or *red dot*. A seen or Yes response to a node branches upward in the tree (to a higher decibel value) and an unseen or No response branches down in the tree (to a lower decibel value). A node with no branches is a leaf and represents the final threshold value that would be returned if that point in the tree is reached. *Black nodes* are for the procedure with a stopping standard deviation of 2.0 dB, and *red nodes* are those with a stopping standard deviation of 2.5 dB. *Green diamonds* indicate decibel values that can be achieved as final threshold values (leaves in the tree), using the *black nodes* but not the *red nodes*. Note that the slight offsets in the vertical direction are for visual clarity; all decibel values at each node are whole integers.

final thresholds that would be possible if a No response was recorded at this location.

3. ROPT_Close is the ROPT of all the ROPT_Yes and ROPT_No ranges of the neighboring locations at this stage in the test giving the range of the possible final thresholds of all the neighbors of the location under consideration.

After each stimulus presentation using the Underlying test, ARBON will check two rules repeatedly at all locations until no action is taken:

- Check Rule 1. If ROPT_Yes is a subset of ROPT_Close and ROPT_No does not overlap either ROPT_Yes or ROPT_Close,

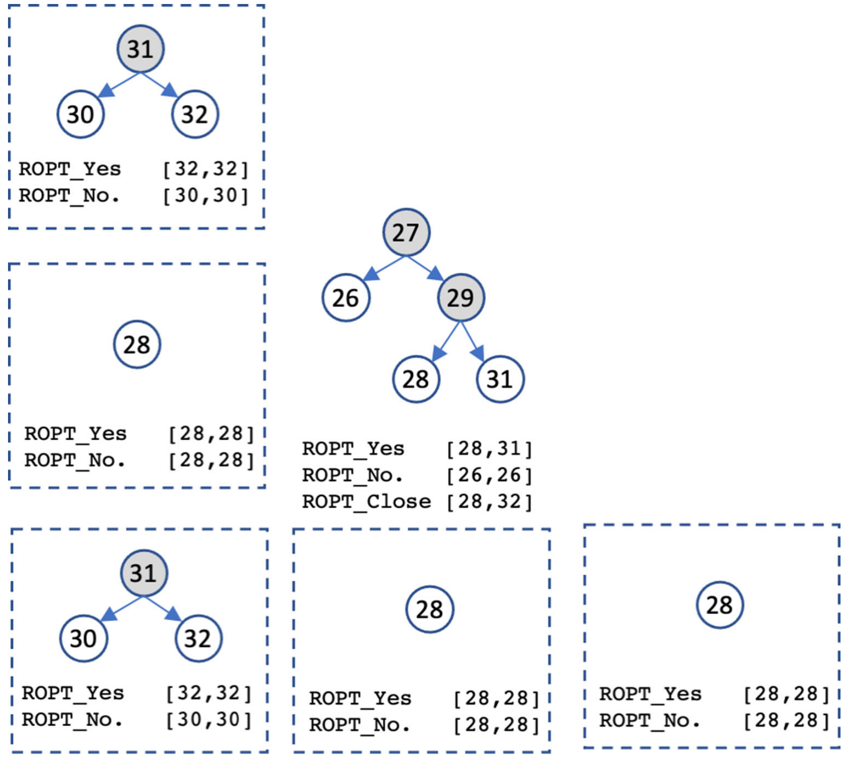

Figure 2. Example calculation of the ROPT of all neighbors (ROPT_Close) at a location in the visual field with five neighbors (dashed boxes). Each location shows the decision tree remaining at that location at this point in the test with *gray nodes* showing possible stimulus presentation values and *white nodes* showing final threshold values. A No response is to the *left*, and a Yes response is to the *right*. The unboxed location has just received a presentation and computes ROPT_Close as the ROPT of the *white values* of all five neighbors. In this case, the location will get an artificial Yes response as Check Rule 1 is triggered (see text).

then inject an artificial Yes response at this location.

- Check Rule 2. If ROPT_No is a subset of ROPT_Close and ROPT_Yes does not overlap ROPT_No or ROPT_Close, then inject an artificial No response at this location.

The assumption behind these rules is that the current location is likely to have a final threshold value close to its neighbors, so if either the Yes or No branch from the current node has a range of possible final thresholds that is completely disjoint from the neighbor's and the other possible thresholds of the other branch, we can exclude it automatically without presenting the stimulus at the current node for a response.

As an example (illustrated in Fig. 2), consider a location in the visual field that has had four presentations already to get to node (4, 27) in Figure 1 (response sequence: Yes to 25 dB, No to 29 dB, No to 27 dB, Yes to 26 dB). For this position in the tree, ROPT_Yes = [28, 31] and ROPT_No = [26, 26]. If all of the neighboring locations have had some presentations resulting in either Yes–Yes or Yes–No–Yes responses—nodes (2, 31) and (3, 28), respectively—then ROPT_Close = [28, 32]. That is, the neighbors can only end up with thresholds in the range of 28 to 32. In this case, ROPT_Yes is a subset of ROPT_Close, and ROPT_No does not overlap ROPT_Yes or ROPT_Close; thus, ARBON assumes a Yes response to 27 dB, skips its presentation, and moves to node (5, 29) in the tree. Note that, if all of the neighboring locations had Yes–Yes responses thus far in the test, then the rule would be triggered again as ROPT_Close is [30, 32], ROPT_Yes is now [31, 31], and ROPT_No [28, 28] and a Yes response would be assumed.

There are two subtleties to this simple approach. First, the order in which locations are tested can alter the triggering of the rules for the same eye. In the experiments in this paper, we chose the next location for presentation during a test randomly from the locations with the lowest presentation count thus far (which is recorded during the test for each location). Second, the experiments in this paper assumed white-on-white perimetry on a 24-2 test pattern which has test locations spaced on a rectangular 6° grid.[1] For this pattern, there can be about a 1 dB difference in thresholds between adjacent neighbors simply due to the eccentricity of the test locations relative to each other. Thus, when making the comparisons in the Check Rules, we adjusted thresholds by an appropriate factor before being aggregated to form the ROPT values. This factor was taken from a Hill of Vision model in figure 12 of Pricking et al.[17]

Although the two check rules introduced above call for strict non-overlapping ranges and subsets for ROPT_Yes, ROPT_No, and ROPT_Close, there is scope for relaxing this precision and allowing a little fuzziness at the boundaries of the ranges. That is, we could allow a little overlap (call it *delta* dB) in the ROPTs, or expansion of supersets to satisfy the ARBON rules for injecting artificial responses. By increasing *delta*, we get more artificial responses and thus a faster test, but threshold values are smoothed to be more like their neighbors. In the experiments discussed below, we used a *delta* value of 1.0 dB.

Data and Implementation

In the experiments described below, we compared three algorithms using computer simulation, which mimics the behavior of a human observer undertaking visual field testing. The three algorithms are

1. Underlying ZEST, which uses the decision tree in Figure 1 (black nodes) at each location in the field
2. Truncated ZEST, which uses the decision tree in Figure 1 (red nodes) at each location in the field
3. Underlying with ARBON added, as just described

Two datasets are used within the computer simulations to simulate performance for observers with normal vision and people with glaucoma. For the normal input dataset, the normal eye dataset is generated from a model of normal white-on-white perimetric thresholds as described by the equations provided in figure 12 of Pricking,[17] which gives a single decibel value at any location in the visual field for an eye of a particular age. To generate threshold values for an eye of a given age, we first choose a general height in the form of probability q between 0.0001 and 0.9999 then take the qth quantile from the normal distributions with means given by the model at each location and standard deviation of 1. To these sampled values at each location, we add a small perturbation (uniform random in the range of -0.5 to 0.5) and finally round the value to a whole decibel. This approach preserves the general shape of the normal hill of vision but introduces fluctuations for the whole eye (q value) and some individual location differences (± 0.5 dB before rounding). Using this approach, we generated 610 normal eyes with the 24-2 pattern, 10 of each age from 20 to 80 inclusive.

The glaucomatous input dataset is a series of 24-2 visual field data collected from 163 eyes with known glaucoma. This dataset has been used in similar visual field procedure computer simulation studies and has been described previously.[4] Visual field damage in this dataset ranged from mild to severe visual field damage (median mean deviation [MD] = -1.81 dB; 5th percentile = +2.14 dB; 95th percentile = -22.55 dB).

Both datasets contain 24-2 test pattern visual field data. For this test pattern, neighboring locations are defined as the nearest test locations (maximum of 8 neighbors) but not crossing either the horizontal or vertical midline (that is, limited to a quadrant). Simulated responses to stimuli are achieved using the SimHenson model of the Open Perimetry Interface.[18] In this model, the probability of seeing a stimulus of x dB is given by

$$p + (1 - p - n)\,(1 - \Phi\,(x, t, \min(6,\ \exp(3.27 - 0.081t)))$$

where p and n are the false-positive and false-negative rates, respectively; t is the assumed true threshold of the simulated location, and $\Phi(x, m, s)$ is the cumulative normal distribution with mean m and standard deviation s. We simulated with two error conditions: reliable, where $p = n = 0\%$, and typical, where $p = 10\%$ and $n = 3\%$.

Data Analysis

For comparison between the procedures, we use the absolute error between the measured threshold and true threshold that is input to the simulation and the number of presentations required at each location across all locations and eyes. This demonstrates the trade-off between accuracy and test time when shortening the procedure using either the Truncated or ARBON approach. As simulated responses are stochastic, and the testing order of locations can alter the performance of ARBON, we repeated each simulation 1000 times and looked at means over the 1000 repeats along with the 95% confidence intervals of the means.

In addition to looking at the average trade-off between speed and error, we also report on the distribution of measured thresholds returned by each procedure at each location over all eyes. In particular, we report the overlap coefficient (OVL) of the frequency distribution of measured thresholds between tests for all eyes for a test run. OVL is simply the proportion of shared area between two histograms of data as exemplified in Figure 3, which can be computed as the

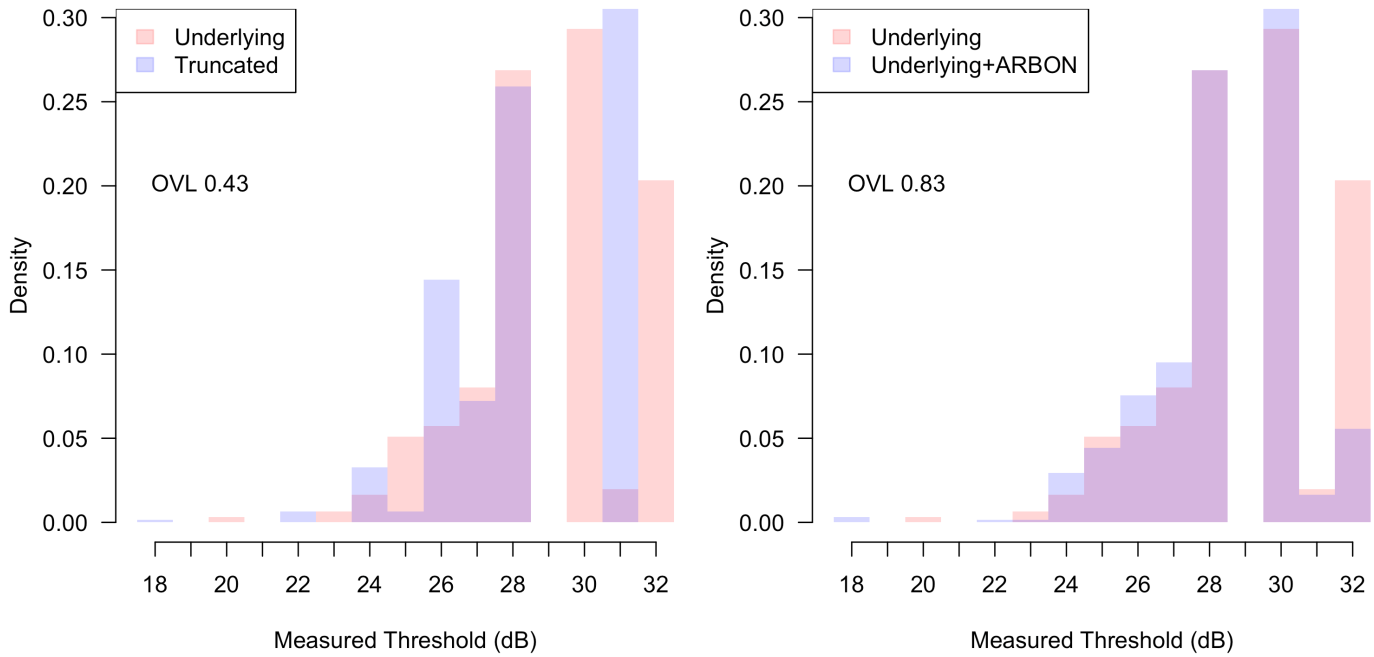

Figure 3. Example of OVL computation for the first run of location (−9, 9) on the Normal 10%/3% dataset as the sum of the dark, overlapping areas of the *red* and *blue histograms* in each case (overlapping areas are indicated by *purple*). The *y*-axis is the normalized frequency count (or probability density) of each measured threshold.

sum of the minimum density of the two procedures for each possible measured threshold value. Thus, 0 represents no overlap, and 1 represents complete agreement. Note that we did not perform conventional inferential statistical testing, because the number of samples in simulation experiments is high, leading to small, clinically meaningless effects becoming statistically significant with sufficient trials.

Results

Figure 4 shows the number of presentations and error for the Underlying, Underlying with ARBON added, and Truncated procedures. As can be seen, adding ARBON decreased the number of presentations by 16% in Normal and 12% in Glaucoma,

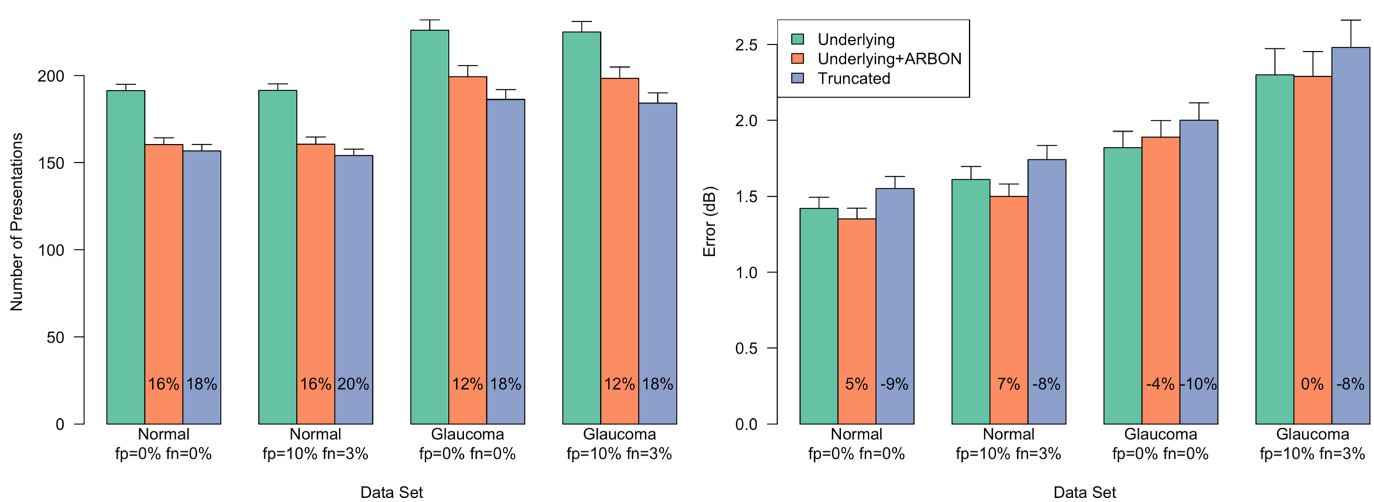

Figure 4. Mean number of presentations and mean absolute error across 1000 repeats of all locations in all eyes for the three procedures for the Normal and Glaucoma datasets with varying false-positive (fp) and false-negative (fn) rates. *Error bars* are 95% confidence intervals of the mean. Percentages show decreases in the mean from Underlying.

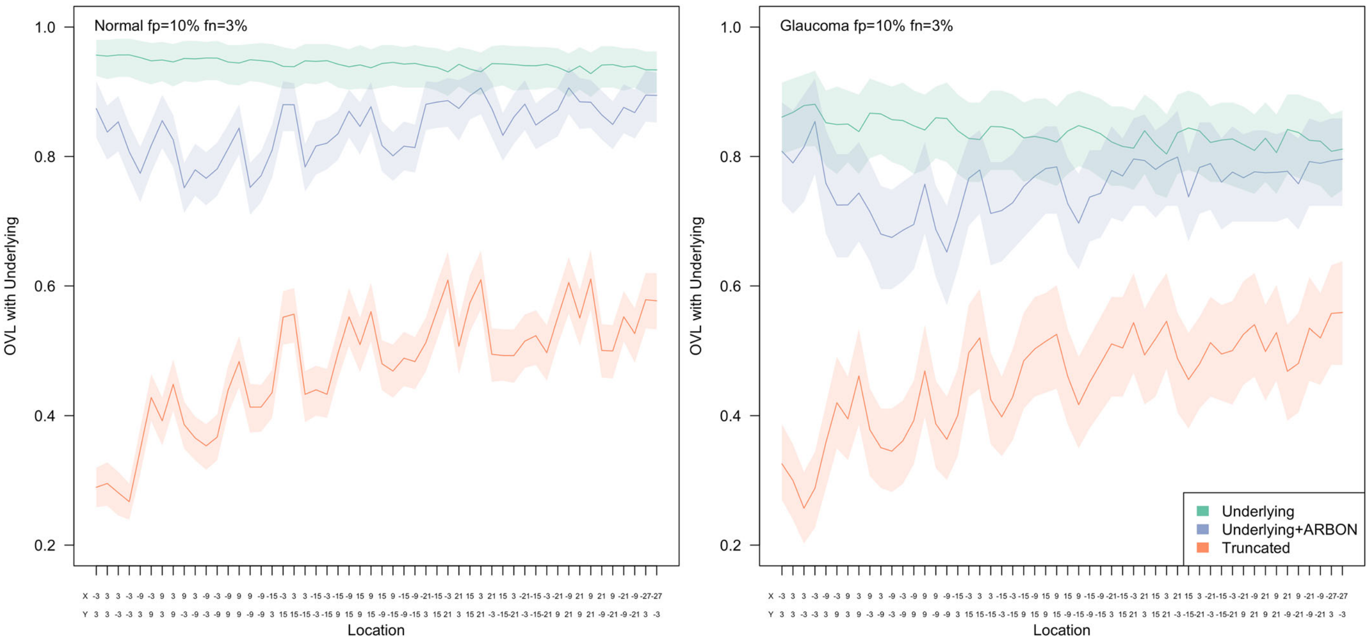

Figure 5. Overlap coefficient OVL for the 52 test locations (sorted by eccentricity on the *x*-axis) for the three test procedures on the two datasets with false responses (Normal $n = 610$, Glaucoma $n = 163$). The *solid line* represents the mean, and the *shading* represents the 95% range of 1000 runs.

whereas Truncated achieved 20% and 18% reductions, respectively. The increased speed came with a corresponding decrease in accuracy of between 8% and 10% for Truncated, but interestingly ARBON showed some small improvements in accuracy in three of the datasets due to smoothing of thresholds introduced by the artificial responses. It is worth noting that the differences in accuracy, averaged across all eyes and all locations, are small in terms of units of decibels and unlikely clinically meaningful.

Figure 5 investigates the distributions of the measured thresholds using each procedure on a location-specific basis (rather than averaged across the visual field). It shows the OVL between Underlying, itself (in green, representing the test–retest OVL of measured thresholds), and the other procedures for 1000 repeats over the 52 locations. As can be seen, Underlying+ARBON had a distribution of measured thresholds much closer to that of Underlying than Truncated; that is, it had a higher OVL. The results for the datasets with no errors are very similar and so are not shown.

Discussion

Using computer simulation is a common method for exploring new perimetric algorithms and provides evidence to support subsequent human testing. Here, we have employed simulation to show that ARBON offers a new way to reduce the number of presentations in a testing algorithm without altering the testing algorithm logic or parameters. The advantage of ARBON over re-engineering the underlying algorithm is that the resulting distribution of measured thresholds returned remains very similar to the underlying procedure, unlike when the underlying procedure is altered either through parameter choices or by alterations to the logic of the procedure.

Given that ARBON uses the same decision tree as the underlying procedure, one would expect the distributions of measured thresholds returned by Underlying and Underlying+ARBON to be almost the same; that is, they would have an OVL close to 1. Figure 5, however, shows that the OVL was closer to 0.8 in these experiments. This difference is almost fully explained by an "extreme value" quirk in the way ARBON has been implemented here to make decisions when a patient responds all No or all Yes for the Underlying procedure. For example, in the Underlying tree, after two Yes responses we ended up at node (2, 31) (Fig. 1), and from here it is possible that with eccentricity corrections and neighbor values all of 32 dB that ARBON injected a No response resulting in an incorrect threshold of 30. Similarly, ARBON often injects a Yes response from node (3,1). With some alterations to ARBON to prevent these two artificial responses, the OVL between Underlying and Underlying+ARBON was about 1.0, but also the number of actual presentations increased. This "extreme value" quirk also explains the upward trajectory of the OVL between Truncated and Underlying. Near the center of the field, threshold values are high, and Truncated often returned its maximum (31 dB) as did Underlying (32 dB), values that do not overlap. As eccentricity increased, the true thresholds decreased, and this "extreme" or "ceiling" effect did not occur as often. It is possible to alter the eccentricity corrections to prevent these effects, but we did not want to over-engineer ARBON for the particular Underlying ZEST in this paper. Instead, these can be used to illustrate aspects of the approach that could be modified on a case-by-case basis to achieve the best outcomes depending on the specifics of the Underlying procedure. Note that we are not advocating that the illustrative Underlying ZEST used in this paper could be used in practice without further engineering.

Another subtle source of difference between measured thresholds of Underlying and Underlying+ARBON is probabilistic. Although ARBON does not alter the decision tree of its underlying procedure, the introduction of artificial responses does somehow alter the procedure in a probabilistic sense. For example, in an underlying procedure, when a stimulus value equal to the subject's threshold is presented, there is a 50% chance of saying Yes and a 50% chance of saying No, assuming no false responses. If ARBON skips this value by injecting its own artificial response based on neighbors, it removes this choice and commits to either Yes or No with 100% certainty. This can lead to some bias in normal threshold values. If it is important that the distribution of measured thresholds is very similar to the underlying procedure (for example, to replicate a normative database collected with the underlying procedure), then it is possible to apply a probabilistic, post hoc correction to measured thresholds that have been obtained with an ARBON response so that the obtained measured thresholds have a distribution much closer to the underlying baseline procedure (see Appendix).

Although we have presented ARBON here in the context of perimetry, it can be added to any psychophysical testing procedure and could even be extended to forced-choice types of procedures. However, for the automatic responses to be triggered, it is necessary for the Yes, No, and Close Ranges of Possible Thresholds to overlap or be disjoint in the prescribed way. For procedures that incorporate a lot of steps to allow for recovery from a false response, this may occur very rarely. For example, in Figure 1, from node (3, 24) it is still possible to have a sequence of No responses that leads all the way down the tree to 0 dB. A location that is in this part of the tree is unlikely to benefit from the logic of ARBON. Similarly, lengthy staircase procedures have possible sets that encompass the entire dynamic range of the instrument for many presentations. Thus, we would only recommend trying ARBON for procedures like those in perimetry that trade off accuracy for speed. Note also that the definition of a neighbor could be extended to include temporal or other forms of "neighbor" relationships within ARBON.

Other methods of exploiting spatial correlations to increase the speed of visual field testing have been explored previously. A simple method of incorporating spatial correlations in a visual field test is to seed the starting point of a procedure at a location with the result of neighboring locations. The assumption is that neighbors are highly correlated. Commonly used perimetric algorithms that use this approach include the Full Threshold algorithm and the SITA family of procedures.[7] These commence by testing four primary locations, one in each quadrant. One consequence of this approach is that locations must have their threshold determination completed before related locations can begin any presentations. An alternative approach is to propagate the response to each stimulus to its neighboring locations, perhaps with weighting factor such as in the SWeLZ[19] or TOP[12] approaches. Both of these approaches embed the spatial logic as an integral component of the testing procedure; hence, any tweaking of parameters or alteration of the logic effectively creates a variant of the procedure that may result in a different distribution of threshold values being returned.

Finally, it is important to note that reducing the number of test presentations is only one contribution to shortening the time of test procedures. Other approaches such as tailoring the time interval for a response to the individual patient (in a Yes–No procedure) or removing additional presentations to check for response reliability (for example, fixation checks, false-positive catch trials) can significantly reduce the overall test time of a test procedure.

In summary, ARBON is an approach that introduces smoothing over neighbors at a presentation level to existing psychophysical procedures without altering the underlying logic or parameters of that procedure. In the case of perimetry, it speeds up a test and provides a distribution of measured thresholds on a population that is close to the original test. The rules of ARBON do not require a certain ordering of test locations which increases the options for choosing locations that are spatially disparate during a test.

Acknowledgments

Disclosure: **A. Turpin**, iCare Finland OY (F); **A.M. McKendrick**, iCare Finland OY (F)

References

1. Anderson DR, Patella VM. *Automated static perimetry*. 2nd ed. St. Louis, MO: Mosby; 1999.
2. Weber J, Klimaschka T. Test time and efficiency of the dynamic strategy in glaucoma perimetry. *Ger J Ophthalmol*. 1995;4:25–31.
3. Luithardt AF, Meisner C, Monhart M, Krapp E, Mast A, Schiefer U. Validation of a new static perimetric thresholding strategy (GATE). *Br J Ophthalmol*. 2015;99:11–15.
4. Turpin A, McKendrick AM, Johnson CA, Vingrys AJ. Properties of perimetric threshold estimates from full threshold, ZEST, and SITA-like strategies, as determined by computer simulation. *Invest Ophthalmol Vis Sci*. 2003;43:4787–4795.
5. Montesano G, Bryan SR, Crabb DP, et al. A comparison between the Compass fundus perimeter and the Humphrey Field Analyzer. *Ophthalmology*. 2019;126:242–251.
6. Matsumoto C, Yamao S, Nomoto H, et al. Visual field testing with head-mounted perimeter “imo”. *PLoS One*. 2016;11:e0161974.
7. Heijl A, Patella VM, Chong LX, et al. A new SITA perimetric threshold testing algorithm: construction and a multicenter clinical study. *Am J Ophthalmol*. 2019;198:154–165.
8. Bengtsson B. A new rapid algorithm for short-wavelength automated perimetry. *Invest Ophthalmol Vis Sci*. 2003;44:1388–1394.
9. Bengtsson B, Heijl A. SITA Fast, a new rapid perimetric threshold test: description of methods and evaluation in patients with manifest and suspect glaucoma. *Acta Ophthalmol Scand*. 1998;76:431–437.
10. Bengtsson B, Olsson J, Heijl A, Rootzen H. A new generation of algorithms for computerized threshold perimetry. *Acta Ophthalmol Scand*. 1997;75:368–375.
11. Kucur Ş S, Márquez-Neila P, Abegg M, Sznitman R. Patient-attentive sequential strategy for perimetry-based visual field acquisition. *Med Image Anal*. 2019;54:179–192.
12. González de la Rosa M, Martinez A, Sanchez M, Mesa C, Cordovés L, Losada M. Accuracy of tendency-oriented perimetry with the Octopus 1-2-3 perimeter. In: Wall M, Heijl A, eds. *Perimetry Update. 1996/1997, Proceedings of the XIIth International Perimetric Society Meeting, Würzburg, Germany, June 4–8, 1996*. Amsterdam: Kugler Publications; 1997:119–123.
13. Artes PH, Iwase A, Ohno Y, Kitazawa Y, Chauhan BC. Properties of perimetric threshold estimates from Full Threshold, SITA Standard, and SITA Fast strategies. *Invest Ophthalmol Vis Sci*. 2002;43:2654–2659.
14. Turpin A, Jankovic D, McKendrick AM. Retesting visual fields: utilizing prior information to decrease test-retest variability in glaucoma. *Invest Ophthalmol Vis Sci*. 2007;48:1627–1634.
15. Turpin A, McKendrick AM, Johnson CA, Vingrys AJ. Development of efficient threshold strategies for frequency doubling technology perimetry using computer simulation. *Invest Ophthalmol Vis Sci*. 2002;43:322–331.
16. Anderson AJ, Johnson CA, Fingeret M, et al. Characteristics of the normative database for the Humphrey Matrix Perimeter. *Invest Ophthalmol Vis Sci*. 2005;46:1540–1548.
17. Pricking S. Age-corrected normal differential luminance values for the entire 80° visual field applying three threshold estimating strategies, using the Octopus 900 perimeter. Tübingen: Eberhard-Karls-Universität zu Tübingen; 2010. Dissertation.
18. Turpin A, Artes PH, McKendrick AM. The open perimetry interface: an enabling tool for clinical visual psychophysics. *J Vis*. 2012;12:22.
19. Rubinstein NJ, McKendrick AM, Turpin A. Incorporating spatial models in visual field test procedures. *Transl Vis Sci Technol*. 2016;5:7.

Appendix

ARBON injects artificial decisions throughout a test procedure that commit to either a Yes or No branch of the decision tree, removing the possibility, however unlikely, of following the other branch. For example, in Figure 1 at node (5, 29 dB), if an eye has a true threshold of 29 dB, then by definition of threshold there is a 0.5 chance of getting a measured threshold of 31 dB and a 0.5 chance of 28 dB. However, if ARBON makes an artificial choice at that node based on neighbors, then only one of the outcomes can occur with certainty. In predominantly normal areas of the visual field, ARBON will more than likely inject a Yes response, and so over a population of eyes true thresholds of 29 dB will be measured closer to 31 dB on average rather than to the expected $(0.5 \times 28) + (0.5 \times 31) = 29.5$.

If it is important for a procedure with ARBON to return population averages the same as the underlying procedure, then we can post hoc correct thresholds so the average is maintained. In our example, whenever a threshold of 31 dB is returned that was arrived at by an artificial Yes at node (5, 29), we could actually return 30 dB, say, thus reducing the average over a population. For thresholds of 31 dB that were arrived at without artificial responses, we can leave the value as 31 dB. The exact amount that thresholds should be altered to preserve a population average can be calculated as follows.

Let $\Psi\ (x,\ t)$ be the probability of seeing stimulus t with true threshold t. One could use the same equation as used for the simulations above for Ψ assuming some representative values of p and n. Let T be a decision tree for some location and some test procedure (like that in Fig. 1). For some given true threshold value t, we can compute the probability of following a branch in T from Ψ and thus the probability of arriving at any leaf as the product of the probability of following the branches that lead to the leaf. Thus, we can compute the expected measured threshold (EMT) for true threshold t using tree T as

$$\mathrm{EMT}\,(t,\,T) = \sum\nolimits_{all\ leaves\ in\ the\ tree} \text{Probability of arriving at leaf} \times \text{leaf dB value}$$

If we assume some distribution of true thresholds to form a population, $P(t)$ = probability of the population having true threshold t, then the mean threshold observed for population P for tree T will be

$$M\,(P,\,T) = \sum_{t=0}^{N} P\,(t) \times \mathrm{EMT}\,(t,\,T)$$

When ARBON injects an artificial response, the EMT of the tree changes as one branch becomes labeled with probability 1.0 and the other 0.0. We can compute this revised EMT′ in the same way as EMT but using the revised leaf probabilities. We can then compute M' using EMT′, and the difference between M' and M becomes the correction to apply to the final threshold.

translational vision science & technology