ATUCAPTS: Automated Tests That a User Cannot Pass Twice Simultaneously

Garrett Andersen and **Vincent Conitzer**
Department of Computer Science, Duke University
Durham, NC, USA
{garrett, conitzer}@cs.duke.edu

Abstract

In highly anonymous environments such as the Internet, many applications suffer from the fact that a single user can pose as multiple users. Indeed, presumably many potential applications do not even get off the ground as a result. Consider the example of an online vote. Requiring voters to provide identifying information, to the extent that this is even feasible, can significantly deter participation. On the other hand, not doing so makes it possible for a single individual to vote more than once, so that the result may become almost meaningless. (A quick web search will reveal many examples of Internet polls with bizarre outcomes.) CAPTCHAs may prevent running a program that votes many times, but they do nothing to prevent a single user from voting many times by hand. In this paper, we propose ATUCAPTS (Automated Tests That a User Cannot Pass Twice Simultaneously) as a solution. ATUCAPTS are automatically generated tests such that it is (1) easy for a user to pass one instance, but (2) extremely difficult for a user to pass two instances at the same time. Thus, if it is feasible to require all users to take such a test at the same time, we can verify that no user holds more than one account. We propose a specific class of ATUCAPTS and present the results of a human subjects study to validate that they satisfy the two properties above. We also introduce several theoretical models of how well an attacker might perform and show that these models still allow for good performance on both (1) and (2) with reasonable test lengths.

1 Motivation

It is well known that the potential anonymity that the Internet currently provides is both a blessing and a curse. On the one hand, it allows the Internet, at least in some cases, to serve as a vehicle for free speech by removing the possibility of in-person retaliation or other undesirable consequences. Without the possibility of anonymity, the Internet may also make it too easy for many entities to obtain very detailed models of users, from which they may suffer in various ways (for example, by being price-discriminated against). On the other hand, the anonymity can enable undesirable behavior as well, such as cyberbullying, libel, child pornography, and various other illegal (and/or immoral) activity. Another issue caused by anonymity, which is the one that we address in this paper, is that it can make it easy for a single user to participate in an online activity under multiple accounts. This can cause a variety of problems, including the following.[1]

1. In online voting, a user would be able to vote more than once, thereby obtaining a disproportionate influence.
2. Closely related, when rating (say) a product online, a user would be able to obtain a disproprtionate influence on the aggregate rating by rating multiple times.
3. In online auctions, a seller could simultaneously participate as a buyer and place shill bids.
4. Combinatorial auctions [Cramton *et al.*, 2006], where multiple items are simultaneously for sale, can often be manipulated by a single bidder pretending to be multiple bidders [Yokoo *et al.*, 2001; 2004].
5. A player may obtain a high rating in an online game by beating many fake accounts with its true account.[2]
6. A person may open multiple e-mail accounts and continue to send spam even after some of them get shut down.

CAPTCHAs [von Ahn *et al.*, 2003; 2004] help address some of these by preventing automated account creation. But even an ideal CAPTCHA does not prevent a dedicated single user from creating multiple accounts by hand.

[1] It should be noted that there may be some benefits to allowing a single person multiple accounts; for example, on an online social network, a person who is involved in multiple social groups may prefer to keep them separate from each other by using multiple accounts. Of course, the social network provider can simply provide functionality for keeping them separate (as in, e.g., Google Circles).

[2] Having multiple accounts may also make it difficult to detect more sophisticated manipuations, such as the classic trick of a weak chess player playing against two strong chess players simultaneously, once as White and once as Black, and copying their moves to the other board. One could also attempt to obtain multiple seats at an online poker table and have these accounts collude with each other against the remaining players. Yet another example would be attempting to play the ESP game [von Ahn and Dabbish, 2004], Peekaboom [von Ahn *et al.*, 2006], etc. with oneself.

2 High-level approach

In this paper, we investigate whether it is possible to have our cake and eat it too—that is, allowing each person to stay anonymous while still being able to obtain at most one account. What we have in mind here is not that the user submits identifying information to the provider—say, a social security number—and then the provider safeguards the person's anonymity. Rather, we want the user to be able to remain anonymous to the provider as well.

At this point, it may seem that there is an impossibility result in store. If a person can obtain an account without giving any identifying information, what is to prevent that same person from doing the same thing again to obtain another account? Still, there may be ways to pull it off. Earlier work [Conitzer, 2010] has investigated the possibility of making it hard to get a second account by leaving a recognizable "cognitive mark" on someone who has already obtained an account. Specifically, this work required a user to pass a memory test—e.g., the user is presented a random subset of a fixed set of images of people's faces to remember, and is then tested on the full set by asking which ones she has seen before. The idea is that this test should be more difficult to pass the second time: the second time one has already seen all the faces the first time around, but she has to remember only which faces she has seen the second time. Unfortunately, experimental results on human subjects for existing versions of such tests are not very strong, in part because performance did not decrease sufficiently from one iteration to the next and in part because there was too much variability in performance across subjects.

In this paper, we take a less ambitious but hopefully more effective approach. We aim to design a test that is easy for a person to pass once, but prohibitively difficult to pass twice *at the same time*. For this criterion, it is fine if a person can easily pass the test twice in a row (sequentially), but not in parallel. A test that meets this criterion would meet our needs in cases where it is reasonable to require all users to be present at the same time. For example, one could organize an online vote as follows. Anyone who wishes to vote should show up on the website at 12:00, and then start the (say) 2-minute test. Anyone who passes the test—which must necessarily happen at 12:02—gets one vote. Under these circumstances, in order to get two votes, a person would have to pass the test twice at the same time,[3] which we are assuming is not possible.

This less ambitious approach may be unsatisfactory for some of the motivating examples given above. For example, for the purpose of signing up for e-mail accounts, it is unreasonable to expect everyone to show up at the same time. For such examples, the approach does not solve the problem. Nevertheless, for other motivating examples, the approach may well suffice, as discussed below. Also, the parallel case might serve as a stepping stone to the sequential case.

- If we consider voting on (say) award nominees, depending on the context we may or may not be able to get people to turn out at the same time. E.g., we could hold a vote on "player of the game" immediately after the game ends.
- If we consider an application such as letting people join a particular chat room, an online poker game, chess game, or other type of game, etc., where we don't want a single individual to join multiple times, the methodology would fit quite well because everyone would necessarily be online at the same time. (We can assume games only start every full ten minutes—12:00, 12:10, ...)
- If we consider a company releasing a new product (or tickets to an event, or something else) at 12:00 at a price that will create demand that will outstrip the immediate supply (e.g., for marketing purposes), and wanting to prevent a single individual from buying up many units in order to immediately resell them at higher prices, this would seem to fit the methodology quite well.
- If we consider participation in an experiment run online (say, on MTurk), we would also not want a subject to participate more than once at the same time because their performance (in the experiment) would drop in a way that does not reflect their true abilities. For this purpose the methodology would also fit quite well. (Of course, if there are multiple iterations of the experiment across time, it would not do anything to prevent the subject from participating in multiple iterations.)

Of course, it is essential that none of these tests can be solved by a computer. Moreover, it should not be possible for a computer-assisted user to pass multiple tests in parallel. This can presumably be achieved by integrating CAPTCHAs into the test, but we have not done so in our human subjects experiments (described below), in which it was not feasible for subjects to set up a computer to help them with the test.

3 Detailed approach

The natural approach to achieve our objective seems to be to design a test that requires dedicated and continuous attention, making it difficult to switch from one instance of the test to another without a noticeable decline in performance. We ended up settling on the following design for the test (a screen shot is provided in Figure 1). Multiple word boxes float around on the screen; the user is supposed to track one of them (which one is indicated once, at the beginning of the test). In this box (as well as in the other boxes), a word is displayed, which changes periodically and is sometimes misspelled. The user is supposed to press one button if the word in the tracked box is spelled correctly, and another if the word is spelled incorrectly. We refer to a phase between word changes as a *query*, and the user's response as an *answer*. The user's score is the number of correct answers given, and the user passes if and only if this score exceeds a preset threshold. The idea is to make it extremely difficult to rapidly and successfully switch between two instances of the test because one will lose track of which box one is supposed to track. At the same time, the boxes should not move so quickly that a

[3] In practice, due to network issues, we may have to allow a user to start the test, say, at any time between 12:00 and 12:01. If so, a person might start one test at 12:00 exactly and another at 12:01 exactly, so that the tests would only partially overlap in time. In our experiments below, we do require the subjects to start the tests at exactly the same time, but it appears that it would be of little help to start the tests at very slightly different times.

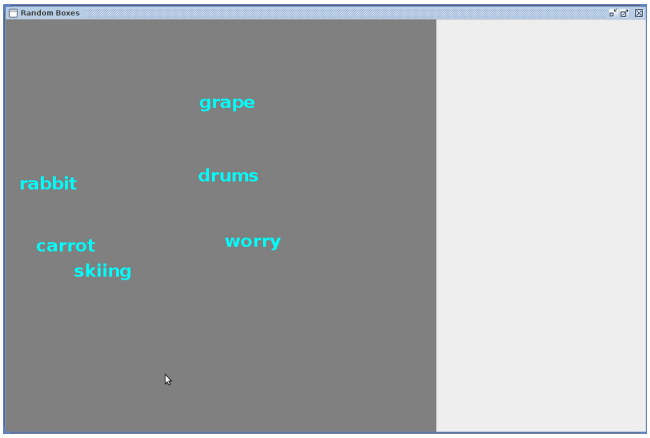

Figure 1: Screenshot. On the left side, the word boxes are moving, and periodically the words in the boxes change. (Note that no actual boxes are visible around the words in our implementation, but it is useful to talk about boxes nonetheless because the words in them change.) The subject has earlier been instructed to pay attention to one specific word box. The right side lights up green on a correct response, and red on an incorrect response. (The right side is made large to make it easy for the subject to see peripherailly.)

user cannot successfully track the box in a single instance of the test (or that it would become too uncomfortable to do so), so we should limit their speed.

Various parameters need to be set to get to a working instantiation of our test, namely the following.

1. The number of boxes. We set this to 6.
2. The speed of the boxes relative to the size of the area in which the boxes move. It would take a box about 3 seconds to traverse the area from top to bottom.
3. The pattern of motion of the boxes. We did the following: for each box, randomly choose a point in the area (not revealed to the user) to which the box moves in a linear fashion. When it reaches that point, randomly choose a new point for it to move to, thereby changing its direction. (This has the advantage that the motion is generally smooth but it is difficult to predict the location of the box a few seconds from now.)
4. The time allocated to each query (i.e., the time interval between word changes). We set this to one second.
5. Feedback to the user during the test. We gave the user immediate feedback upon pressing a button (green for correct, red for incorrect) because not giving any feedback would seem to make it difficult to learn to perform well and also runs the risk of complete failure (for example, if the user is pressing the wrong button). However, we did not give the user cumulative feedback (such as the current score) until the end.

All of these choices (as well as further ones described in the next section) were based either on intuitive judgment or informal preliminary testing on the authors themselves, and not much time was spent optimizing the parameters. Hence, it is likely that other choices within this framework would have performed even better. For example, the middle of the area sometimes becomes quite crowded with word boxes, and it may be preferable to change the distribution of motion accordingly. Even more likely is that there is an entirely different framework that would have performed even better. If so, it only strengthens the potential of our general approach.

4 Related work in the psychology literature

The closest work in the psychology literature of which we are aware concerns split visual attention. This work generally involves subjects being in front of a visual display, which displays visual cues (e.g., boxes) as well as stimuli (e.g., a dot appearing inside or outside of the boxes). Most research has focused on whether it is possible for a subject to simultaneously pay attention to multiple regions that are not contiguous. Proponents of the view that attention is unitary and indivisble [Eriksen and Yeh, 1985; McCormick and Klein, 1990; Pan and Eriksen, 1993; McCormick *et al.*, 1998; Jans *et al.*, 2010] must explain how subjects attain some degree of success on these tests. Two explanations that have been proposed are that attention shifts back and forth serially [Eriksen and Eriksen, 1974; Posner, 1980] and that the area of attention can be expanded to encompass the relevant region [Eriksen and James, 1986]. Others support the view that attention can be genuinely split across multiple noncontiguous regions [LaBerge and Brown, 1989; Castiello and Umiltà, 1992; Cave *et al.*, 2010], at least under certain circumstances [Lim and Lee, 2011; Yap and Lim, 2013].

What the subjects are asked to do in this work is quite different from what they are asked to do in our setup; notably, the stimuli in this previous work are much simpler in nature. More generally, we note that the models of (visual) attention that this previous work concerns are neither necessary nor sufficient for our purposes, as we discuss below. (This is of course not to dispute their value for other purposes!)

They are not necessary insofar as they are designed to characterize the boundary between what human attention is and is not capable of. (Moreover, this boundary might vary from person to person, so that really a statistical characterization is needed.) In contrast, for our purposes, it is not necessary to find the exact boundary; all we need is to design a test that is far on the positive side of the boundary when taken once, and far on the negative side when taken in duplicate.

They are also not sufficient insofar as they are not designed for the adversarial conditions of our application. They only attempt to characterize a normal human being acting in accordance with the rules of the experiments. In contrast, in our setting we have to consider more adversarial agents who might adopt various high-level strategies to defeat the test. For example, some of the work discussed above involves a person having to pay attention to a large visual space and notice occurrences in it. Such a design would not work well for our purposes; for example, taking such a test at home, a person could rescale the image area to make it occupy a smaller part of her visual space, or even make the two places with occurrences spatially coincide. We need a design that fits our

adversarial environment better. This is what motivates us using moving boxes and a task that cannot be well accomplished with peripheral vision. The key difficulty for us is not a subject that is truly simultaneously watching both instances of the test (as in the experimental setups in the above work), but rather one that tries to rapidly switch back and forth between them. This is what motivates our decoy boxes.

5 Evaluation on human subjects

We now present the results of our human subjects study based on the detailed approach in Section 3. 25 subjects were recruited by posting on an internal university website/list (a "free classifieds marketplace"). Each subject received $10 for showing up, in addition to rewards described below. For each subject, the study consisted of two phases. In phase 1, they were asked to complete a single copy of our test, with 80 queries and a predetermined threshold of 64 for passing. They were promised a reward of US $5 for passing the test. In phase 2, they were asked to complete two copies of our test in parallel. They were promised a reward of US $10 for passing *both* tests (but nothing for passing just one of them, to incentivize them to try to pass both). Before each phase, subjects were allowed as many practice runs as they liked. The table in Figure 2 shows the results. We also show the number of practice runs subjects took in the first phase. (We do not show this for the second phase, because many subjects would frequently restart the practice run, making it challenging to assess what counts as a practice run.)

subject number	#practice runs (in first phase)	score on single test	scores on simultaneous tests
1	1	70	41; 34
2	1	73	61; 52
3	1	71	62; 41
4	3	74	32; 11
5	1	76	31; 32
6	2	72	49; 29
7	1	77	35; 35
8	1	75	31; 57
9	1	74	24; 25
10	2	75	27; 31
11	2	71	25; 29
12	2	73	38; 21
13	1	77	22; 27
14	1	71	25; 20
15	1	75	58; 34
16	1	76	57; 26
17	1	76	41; 35
18	1	76	33; 27
19	2	72	26; 31
20	1	76	72; 46
21	1	68	46; 34
22	1	71	40; 56
23	1	76	70; 35
24	1	76	39; 35
25	1	67	33; 26

Figure 2: Results of human subjects experiment.

The results are encouraging. Every subject passed the single test. (Scores on the single test varied from 67 to 77, with an average of 73.5.) No subject passed *both* of the simultaneous tests. The closest to passing both were subjects number 2 (with the highest lower score, 52) and 20 (with the highest combined score, 118). (The lower score varied from 11 to 52, with an average of 30.7; the higher score varied from 25 to 72, with an average of 43.2. Note that the lower score is more relevant for our purposes.) Moreover, the generally low numbers of practice runs in the first phase suggest the test does not pose a high barrier (for this demographic).

6 Theoretical models

While our experimental results were encouraging, one may worry that users with more time to prepare and more motivation would have some significant probability of passing the test twice in parallel, both by adopting better strategies and by developing skill at these strategies. In this case, it is possible that we can still prevent them from passing the test twice in parallel by adjusting the test. Of course we can simply increase the threshold, but this will eventually exclude legitimate users. On the other hand, increasing the length of the test (and thereby the number of answers given) will reduce the variance in the percentage of correct answers, for both legitimate users and attackers. Hence, it will make legitimate users more likely to succeed and attackers more likely to fail. But how does the length of the test relate to these success rates? In this section, we develop some theoretical models to get a basic sense of this.

It should be noted that the purpose of these models is different from that of models that one might find in the psychology literature, because our overall goal is different. It is not to accurately predict the performance of a person on reasonable tests across a range of conditions. Instead, we try to make the conditions of the test extreme enough that there is no way to pass more than one instance of the test in a straightforward way. Our models, rather, aim to bound how well a person could do with different high-level strategies, going back and forth between the tests in some particular way. Hence our theoretical models are really not about giving us insight into the attention capabilities of the human brain, but rather about what various kinds of strategizing could achieve.

The following assumptions are common to all our models.

- Whether or not a legitimate user (taking a single test) responds correctly to a given query is a random variable, and these random variables are drawn i.i.d. In particular, this implies that the length of the test does not affect the rate of correct responses; it would be good to, in future research, account for users getting better due to learning or worse due to fatigue.
- An attacker (taking two tests simultaneously) can pay attention to at most half the queries across both tests. He responds correctly to 100% of the queries to which he pays attention. He responds correctly to 50% of the queries to which he is not paying attention (i.i.d.).

For each of the specific models that we are about to introduce, the following quantities are related:

- p: (a lower bound on) the probability with which a legitimate user answers a given query correctly,
- n: the number of queries in the test,
- t: the threshold number of queries one needs to answer correctly to pass the test ($t \leq n$),
- λ: the minimum probability with which a legitimate user should pass the test, and
- α: the maximum probability with which an attacker should pass both tests.

We take p to be given exogenously by human cognitive architecture; a reasonable estimate based on our experiments above would be 0.9. Then, for any combination of n, λ, and α (which values for these are reasonable will depend on the application), we can ask whether there exists a value of t such that:

(1) a legitimate user passes with probability at least λ, and

(2) an attacker succeeds with probability at most α.

In any one of our models, (1) holds if and only if $1 - F_B(t; n, p) \geq \lambda$, where F_B is the cumulative probability distribution (CDF) of the binomial distribution, i.e., $F_B(t; n, p) = \sum_{k=0}^{t-1} \binom{n}{k} p^k (1-p)^{n-k}$. The condition for whether (2) holds depends on the behavior of the attacker. We now introduce three models, ranging from a weak attacker to a strong attacker.

Model 1. In this model, we assume that the attacker pays full attention to one test (which he will definitely pass), and guesses randomly on every single query on the other test. Thus, (2) holds if and only if the probability of passing the test by random guessing is low enough, i.e., $1 - F_B(t; n, 1/2) \leq \alpha$. This is a very weak model of an attacker; on the other hand, this is the strongest attack that subjects in our experiment managed to perform (with the vast majority of subjects just guessing randomly on *both* tests).

Model 2. In this model, we assume that the attacker pays attention to each of the two tests exactly half the time. Hence, the attacker will certainly get $n/2$ responses right on each test, and the cumulative probability distribution for how many of the remaining $n/2$ responses he gets right is $F_B(\cdot; n/2, 1/2)$ (and he needs to get an additional $t - n/2$ right to pass it). Thus, (2) holds if and only if $(1 - F_B(t - n/2; n/2, 1/2))^2 \leq \alpha$.

Before we introduce Model 3, we first give a natural *upper bound* on how well the attacker can do, assuming that he can pay attention to at most half the queries across the two tests and at best guess randomly on the other ones. This upper bound can be interpreted as the relaxation (unrealistic model) where the attacker can *after the fact* reallocate correct responses from one test to the other. Then, we show that Model 3 actually achieves this upper bound.

Upper bound 1. The attacker needs to get at least $2t$ correct responses from the $2n$ queries across the two tests. He will certainly get n responses right, and the probability distribution for how many of the remaining n responses he gets right is $F_B(\cdot; n, 1/2)$ (and he needs to get an additional $2t - n$ right to pass both). Thus (2) holds if and only if $(1 - F_B(2t - n; n, 1/2)) \leq \alpha$.

Model 3. In this final model, we assume that the attacker pays attention to the test in which he has the lower score so far (choosing arbitrarily if the scores are the same).

Proposition 1 *In Model 3, the attacker succeeds if and only if his total number of correct answers is at least* $2t$*. Therefore, as in Upper Bound 1, (2) holds if and only if* $(1 - F_B(2t - n; n, 1/2)) \leq \alpha$.

Proof: We prove the first sentence of the proposition, from which the second immediately follows. Under Model 3, at any point in time, the attacker's score in one test must always be within 1 of the attacker's score on the other test, because of the following proof by induction on the number of queries asked in each test so far. Initially (0 queries asked) this is true. If it is necessarily true when k queries have been asked, then it is also necessarily true when $k + 1$ queries have been asked. This is because before the $(k + 1)$th query (right after the kth), either (a) the two scores were the same, in which case they are certainly within 1 of each other one query later; or (b) (without loss of generality) the score on test 1 was one lower than the score on test 2 after the kth query, in which case the attacker pays attention to test 1 next, so that its score will advance by 1 and test 2's score advances by at most one. Then, if the attacker gets at least $2t$ reponses right, it cannot be the case that his score on one test is at most $t - 1$, because the score on the other test would have to be at least $t + 1$, contradicting that they are within 1 of each other. Conversely, if the attacker gets fewer than $2t$ responses right, clearly he cannot pass both tests. ∎

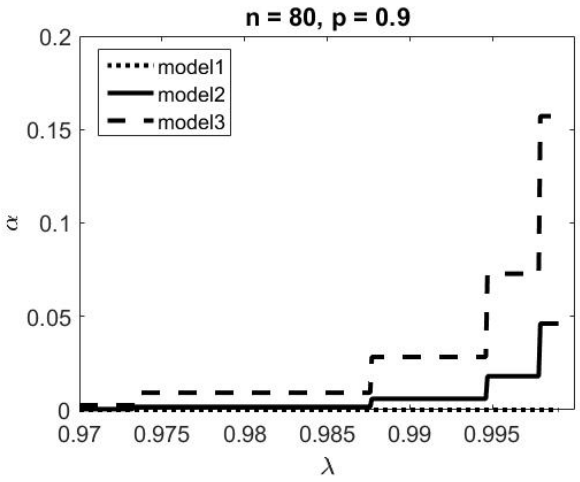

Figure 3: Optimal combinations of α and λ (Pareto frontier) achievable with some choice of t, holding n (as well as p) fixed. (Combinations to the northwest are also feasible.)

Now that we have our three models, we can, for each of them, determine which combinations of n, λ, and α are feasible (holding p fixed at 0.9, a bit below the fraction of correct responses in our study for a single test, and being able to choose any t). Figures 3, 4, and 5 illustrate which combinations are feasible. Each figure fixes one of n, λ, and α, and

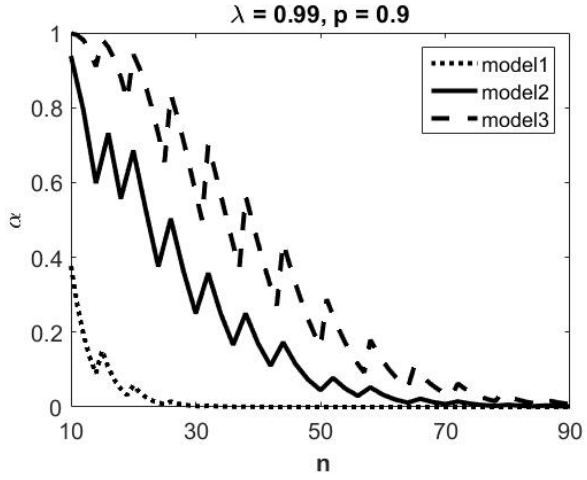

Figure 4: Optimal combinations of α and n (Pareto frontier) achievable with some choice of t, holding λ (as well as p) fixed. (Combinations to the northeast are also feasible.)

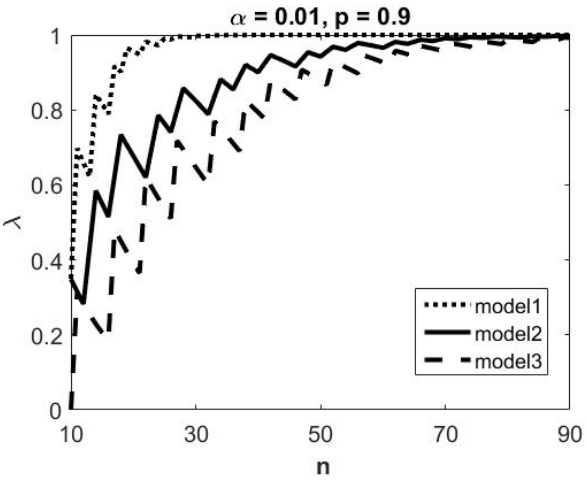

Figure 5: Optimal combinations of λ and n (Pareto frontier) achievable with some choice of t, holding α (as well as p) fixed. (Combinations to the southeast are also feasible.)

shows which combinations of the remaining two are feasible under each model. For example, if we require $n = 80$ and we use Model 3, then we can achieve $\alpha = .0092$ and $\lambda = .98$, or $\alpha = .0283$ and $\lambda = .99$ (Figure 3; note that in this figure, the plot for model 1 has some tiny values of α, because tiny values of α can be obtained even with reasonable settings for the other parameters under this model); if we require $\lambda = .99$ and we use Model 3, then we can achieve $\alpha = .1225$ using $n = 60$ or $\alpha = .0074$ using $n = 90$ (Figure 4); if we require $\alpha = .01$ and we use Model 3, then we can achieve $\lambda = .9269$ using $n = 60$ or $\lambda = .9925$ using $n = 90$ (Figure 5). The fact that the sequences proceed in jumps (and, in Figures 4 and 5, are non-monotone) is due to the fact that t must be an integer. In Figure 3, as we increase λ, at some point we will need to drop t by 1, causing an upward jump in α. In Figure 4, as we increase n in increments of 1, we can generally increase t in increments of 1 as well, but at some point doing so would push λ below the fixed threshold, so that we need to leave t fixed for one step (while n still increases by 1), causing an upward jump in α. In Figure 5, as we increase n in increments of 1, we generally need to increase t in increments of 1 as well to keep α below the threshold, but at some point we can leave t fixed for one step (while n still increases by 1), enabling an upward jump in λ.

From these figures, we can conclude that even under Model 3, it is possible to achieve reasonable values of α and λ with a reasonable test length (say, $\alpha = .01$, $\lambda = .9925$, $n = 90$).

7 More sophisticated attackers

The models discussed above—even Model 3—still have a number of unstated restrictions on the attacker. For one, they assume that the attacker cannot use a computer to pass (or help him pass) the test. Preventing this could involve integrating CAPTCHAs. For example, the words could be written as CAPTCHAs. In this case, of course, it should be tested whether this affects the performance of a legitimate user. One may wish to integrate CAPTCHAs to an even greater extent to prevent the attacker from having the computer guess randomly on a very large number of instances of the test. If we are not confident that we can effectively prevent such large-scale random guessing, we can at least cancel the vote (or whatever the event is) if we see a very large number of failed attempts to pass the test. A downside of this approach is that it enables a particular kind of denial-of-service attack, where an attacker can cancel the event by participating very often.

Even if we succeed in making computer-aided attacks completely ineffective, we might still imagine a single person trying to pass multiple instances of the test at once by hand. Given that it is already challenging to pass two instances at once, it may not appear to be a good idea for the attacker to split his attention across more than two. But the attacker could, for example, guess randomly on all but one instance. It is not immediately obvious in how many instances of the test someone could feasibly guess randomly. We could make doing so more difficult by adding a small amount of random noise to the timing of when the words change, so that the timing of a person who is simultaneously hitting keys for many instances is detectably off relative to the actual timing of those instances. But perhaps the attacker could succeed in guessing randomly on a few instances. The analysis for Model 1 can be used to extend to this case. Specifically, if a user manages to guess randomly on m instances of the test, the probability that she fails on all of these is $[F_B(t; n, 1/2)]^m$. Figures 6 and 7 show the probability that at least one of these random guessing attempts is successful, as a function of m, holding the other parameters fixed at reasonable values. Figure 6 shows that when m is not extremely large, this probability is extremely low. It increases almost linearly; this is due simply to the fact that the probability of getting at least one success from m Bernoulli trials is roughly linear in m when p and m are small. Figure 7 shows that when m is extremely large—tens to hundreds of millions—the probability of one of them being successful becomes significant (and is no longer almost

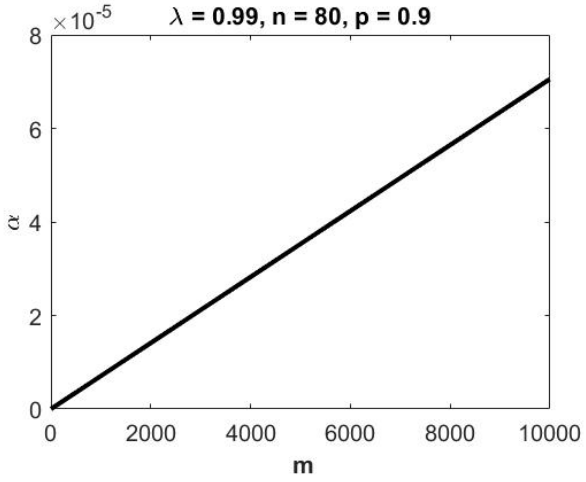

Figure 6: Probability of getting at least one success out of m random guessing attempts as a function of m, holding λ and n (as well as p) fixed, for reasonable values of m.

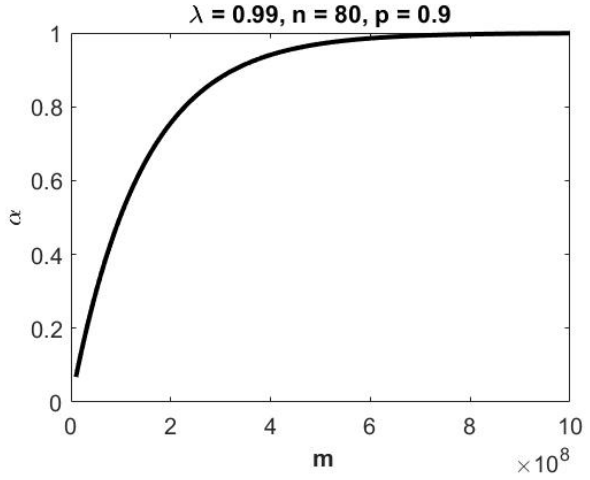

Figure 7: Probability of getting at least one success out of m random guessing attempts as a function of m, holding λ and n (as well as p) fixed, for extremely large values of m.

linear). Such large values of m do not seem to be a signficant concern for our purposes.

8 Conclusion

In this paper, we presented a methodology that has the potential to address many problems involving people participating more than once in online events. We require participants to show up at a given time, and ask them to pass a test that is easy to pass once but extremely difficult to pass more than once *at the same time*. We designed and implemented a basic version of such a test and used it to conduct a study on human subjects. Every subject was able to pass the test, and no subject was able to pass two at once. We also introduced theoretical attacker models that involved more sophisticated attacks than we observed from our subjects, and showed that under the assumptions of these models, such tests can still be successful while remaining of a reasonable length. Finally, we discussed even more sophisticated attacks that could be performed if the test were deployed "out in the wild" and discussed how such attacks might be addressed. Likely, there are other possible attacks that we have not anticipated, and it would seem unwise at this point to rely on this methodology to secure a high-value event. An interesting challenge for future research is to design a way to evaluate how well such a test truly performs in practice, because we would necessarily give up the ability to observe what the attacker is doing, unlike the controlled setting we had in our human subjects study. (Of course, similar issues occur in the deployment of, say, cryptographic protocols.) Another challenge is to create a version of the test that is accessible to populations that would not be able to pass under the current design, e.g., those with sight impairment.

Overall, though, we are encouraged by the experimental results that we obtained with this first design, and imagine that there is still much room for future research to improve on the methodology—for example, reducing the length of the test or addressing more sophisticated attacks. An ever-increasing amount of human activity is moving online, and this enables existing social mechanisms to be implemented more efficiently as well as entirely new ones to be developed. A downside is that many of these mechanisms could be vulnerable to a person participating more than once. This appears to set up a tension between the anonymity of the Internet and the objectives of the mechanisms. However, we believe that the types of techniques discussed in this paper may allow us to sidestep this tradeoff and maintain anonymity without significant loss in the objectives. Clearly, the specific technique introduced in this paper, even if developed further to the point that it is ready for widespread adoption, will help only for mechanisms where it is reasonable to expect participants to all show up at the same time. Nevertheless, this may be a stepping stone towards techniques that solve the problem more generally, for example by solving the sequential variant discussed at the beginning of this paper.

Acknowledgments

We thank Eric Hu for writing code for the software used in the human subjects experiment (described in Section 3), and Kobi Gal and the anonymous reviewers for feedback on the paper. We are also thankful for support from NSF under awards IIS-1527434, CCF-1101659, IIS-0953756, and CCF-1337215, ARO under grants W911NF-12-1-0550 and W911NF-11-1-0332, and a Guggenheim Fellowship. Part of this research was done while Conitzer was visiting the Simons Institute for the Theory of Computing.

References

[Castiello and Umiltà, 1992] Umberto Castiello and Carlo Umiltà. Splitting focal attention. *Journal of Experimental Psychology: Human Perception and Performance*, 18(3):837–48, 1992.

[Cave *et al.*, 2010] Kyle R. Cave, William S. Bush, and Thalia G. Taylor. Split attention as part of a flexible at-

tentional system for complex scenes: comment on Jans, Peters, and De Weerd (2010). *Psychological Review*, 117(2):685–96, 2010.

[Conitzer, 2010] Vincent Conitzer. Using a memory test to limit a user to one account. In Wolfgang Ketter, Han La Poutré, Norman Sadeh, Onn Shehory, and William Walsh, editors, *Agent-Mediated Electronic Commerce and Trading Agent Design and Analysis*, volume 44 of *Lecture Notes in Business Information Processing*, pages 60–72. Springer, 2010.

[Cramton *et al.*, 2006] Peter Cramton, Yoav Shoham, and Richard Steinberg. *Combinatorial Auctions*. MIT Press, 2006.

[Eriksen and Eriksen, 1974] Barbara A. Eriksen and Charles W. Eriksen. Effects of noise letters upon the identification of a target letter in a nonsearch task. *Perception & Psychophysics*, 16(1):143–149, 1974.

[Eriksen and James, 1986] Charles W. Eriksen and James D. St. James. Visual attention within and around the field of focal attention: A zoom lens model. *Perception & Psychophysics*, 40(4):225–240, 1986.

[Eriksen and Yeh, 1985] Charles W. Eriksen and Yei-yu Yeh. Allocation of attention in the visual field. *Journal of Experimental Psychology: Human Perception and Performance*, 11(5):583–97, 1985.

[Jans *et al.*, 2010] Bert Jans, Judith C. Peters, and Peter De Weerd. Visual spatial attention to multiple locations at once: the jury is still out. *Psychological Review*, 117(2):637–84, 2010.

[LaBerge and Brown, 1989] David LaBerge and Vincent Brown. Theory of attentional operations in shape identification. *Psychological Review*, 96(1):101–124, 1989.

[Lim and Lee, 2011] Stephen Wee Hun Lim and Li Neng Lee. When (and why) might visual focal attention split? In *European Perspectives on Cognitive Science*. New Bulgarian University Press, 2011.

[McCormick and Klein, 1990] Peter A. McCormick and Raymond Klein. The spatial distribution of attention during covert visual orienting. *Acta Psychologica*, 75(3):225–242, 1990.

[McCormick *et al.*, 1998] Peter A. McCormick, Raymond M. Klein, and Susan Johnston. Splitting versus sharing focal attention: Comment on Castiello and Umiltà (1992). *Journal of Experimental Psychology: Human Perception and Performance*, 24(1):350–7, 1998.

[Pan and Eriksen, 1993] Kaiyu Pan and Charles W. Eriksen. Attentional distribution in the visual field during same-different judgments as assessed by response competition. *Perception & Psychophysics*, 53(2):134–144, 1993.

[Posner, 1980] Michael I. Posner. Orienting of attention. *Quarterly Journal of Experimental Psychology*, 32(1):3–25, 1980.

[von Ahn and Dabbish, 2004] Luis von Ahn and Laura Dabbish. Labeling images with a computer game. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 319–326, Vienna, Austria, 2004.

[von Ahn *et al.*, 2003] Luis von Ahn, Manuel Blum, Nicholas Hopper, and John Langford. CAPTCHA: Using hard AI problems for security. In *Advances in Cryptology - EUROCRYPT 2003, International Conference on the Theory and Applications of Cryptographic Techniques*, pages 294–311, Warsaw, Poland, 2003.

[von Ahn *et al.*, 2004] Luis von Ahn, Manuel Blum, and John Langford. Telling humans and computers apart automatically: How lazy cryptographers do AI. *Communications of the ACM*, 47(2):56–60, February 2004.

[von Ahn *et al.*, 2006] Luis von Ahn, Ruoran Liu, and Manuel Blum. Peekaboom: a game for locating objects in images. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 55–64, Montréal, Québec, Canada, 2006.

[Yap and Lim, 2013] Jit Yong Yap and Stephen Wee Hun Lim. Splitting visual focal attention? It probably depends on who you are. In *Proceedings of the 2nd Annual International Conference on Cognitive and Behavioral Psychology, Singapore, February*, 2013.

[Yokoo *et al.*, 2001] Makoto Yokoo, Yuko Sakurai, and Shigeo Matsubara. Robust combinatorial auction protocol against false-name bids. *Artificial Intelligence*, 130(2):167–181, 2001.

[Yokoo *et al.*, 2004] Makoto Yokoo, Yuko Sakurai, and Shigeo Matsubara. The effect of false-name bids in combinatorial auctions: New fraud in Internet auctions. *Games and Economic Behavior*, 46(1):174–188, 2004.