Reply to Specific Comments

Editor Comments

> **As you will see from reading the reviews, there are considerable differences in reviewer opinions. I think it is clear that all three think this should be published. But while Reviewers 1 and 2 have rather minor comments/recommendations, Reviewer 3 has a fairly extensive history lesson for you. He/she clearly thinks that there needs to be a better attempt in the PsiTurk paper to connect your work as an extension to the long history of online experimentation systems in psychology. He/she even has a set of slides from a 2008 talk that I will send to your via email. Honestly, I think that 1-2 paragraphs putting PsiTurk in the context of previous online experimentation engines, and appropriately paying homage to our elders would suffice.**

In the introduction of the paper, we now provide a summary of past work on online experimentation systems. It is important to point out that the manuscript we provided is really a summary of the work the open-source community has done on this particular project. The software provides help for running experiments online, particularly using Amazon Mechanical Turk. It provides some libraries to help build experiments but actually provides more in terms of web server architecture, databases, etc.… Thus, while there is a large literature on "good practices" for online experimentation, our paper has slightly different goals. That said, we have taken the reviewer's comments to heart and added references and discussion where appropriate.

> **I invite you to have a look at the reviewers' comments and submit a revision to the manuscript. I hope to see your revised manuscript within 60 days, since any changes that would be made are rather minor. Obviously, just let me know if you need more time. I do not anticipate sending the revision back out for review.**

I apologize for the delay. This manuscript fell through the cracks of my sabbatical.

Reviewer 1

> **Looks good! Just three minor comments you can pass along:**
>
> - **It would be helpful to know what happens if the local server hosting the experiment goes down. Is the local server synced with the ad server to ensure that workers cannot sign up for currently inaccessible HITs? Or can workers sign up for HITs that they are then unable to complete, resulting in the experimenter having to deal with the fallout? If the latter, it would be good to note somewhere in the manuscript that researchers running their experiment locally should probably have a separate machine dedicated to the experiment in order to prevent unexpected downtime.**

If the local server hosting the experiment goes down any participants currently taking the experiment may encounter problems. However, it should be possible in most cases to restart the server and allow participants to continue (assuming the server interruption is temporary). If the server is offline or unreachable, new participants will not be able to begin the experiment. The ad server communicates with the psiTurk researcher's local machine during the display of the ad to verify it is online. It is true that the current psiTurk model encourages researchers to manage their own server (e.g., using a local laptop) and that can have downsides. The upsides are that you don't need to have a server, and make sure you server files are synchronized with the ones you edit on your local computer. We now mention this more explicitly in the section titled Section 7.3 "The Secure Ad Server".

- **It would be nice to see at least a little more technical discussion of the platform--what programming languages and web frameworks it's built on, what motivated those choices, etc. This kind of thing wouldn't matter for many GUI-based packages where the user isn't expected to know any programming, but in this case psiTurk is geared towards people with at least some programming experience, so it makes sense to spell out some of the design choices. It's notable, for example, that the word "Python" doesn't appear in the article until page 7, and Flask (the Python web framework psiTurk is built on) is never mentioned at all.**

This is a great point. We have added information about the major dependencies in “Section 6.1 System requirements” which explores more of the packages that psiTurk depends on and what you need to know about to get started.

- **It's unclear why psiTurk can't be run on Windows; the Anaconda bundle is available for Windows, and in principle anything that runs in pure Python should be executable on Windows as well. If it's just that the authors haven't tested PsiTurk on Windows, they could perhaps update the text to indicate that psiTurk may potentially run on Windows, but that that isn't officially supported. Also, under options for running the code on a Windows platform, they may want to add virtualization as a viable option, since there are now relatively lightweight options for running a linux VM on Windows (e.g. VirtualBox).**

psiTurk uses some special libraries for starting and controlling processes that depend on a UNIX-based environment. In addition, some versions of Windows have security features which make it difficult to run server processes that accept incoming connections. It is possible to run psiTurk in a virtual machine. In addition, many users have found ways to run psiTurk on cloud hosting services like Amazon Web Services or Red Hat’s OpenShift and the documentation guides new users through that process.

Reviewer 2

Summary

The authors describe psiTurk, a framework for conducting and managing studies through Amazon Mechanical Turk. The paper details the infrastructure of the system, the general needs it addresses in the field, and gives some basic examples of its use.

Recommendation

This is a great paper and will no doubt attract great interest. The framework itself is quite exciting and I’m interested in using it myself. I just have a few simple suggestions to improve the manuscript for the readership of BRM.

Thanks the positive comments!

1) Some quick details on the front end (minor expository)

I wondered if the authors could open the paper with a bit more clear detail on what psiTurk accomplishes. As someone who has used AMT extensively, and has done extensive JS programming to create real-time cognitive tasks, I felt at a loss for the first several pages what psiTurk would do for me in concrete specifics. The framework, of course, is really cool and exciting. Maybe just being a bit more explicit about the structure of psiTurk at the front end would

be helpful. For example, the last paragraph of the opening section details some of this, but the paper then jumps into the survey study about what researchers need and use and so on. In that paragraph, it may be good to note the command line at least, since that is such a central part of the system and is crucial for how the user will engage psiTurk. Just laying out the framework a bit more to hint at the tool's tech specs would help.

This is a very good comment and we added some text to the Introduction to help frame the project better. One of the major challenges in creating a software artifact is convincing people who are living their life fine without it they need it. In this case, I think may people are doing online experiments already and may not realize exactly what psiTurk helps with. We tried our best to address this.

2) Stroop / example task

The Stroop task example could be described in a bit more detail for the reader, even if a paragraph. One idea is to create a new section that provides some quick details about how one would get started and setup an actual task — how much do I do in JS, MySQL/SQLite, etc., and how much does psiTurk help me? What are the stages I use to deploy the task online, etc.? The Stroop task would be a great domain in which to describe these steps. I imagine this would be really easy to write, and be just a few paragraphs. It would help with a bit more detail to go along with the "quick start" description that is already there.

Section 7 "Getting started with psiTurk" provides a fairly detailed walk through of running an experiment and how the various pieces connect. We have tried to elaborate that a little bit but feel that the online documentation may provide a better avenue for detailed step-by-step instructions. The purpose of this paper is to explain the overall architecture and goals of the project.

3) Flat-file saving on the web

As just a small critique, the server automatically manages unique identifiers that can be employed for ensuring unique data files when saving. For example, session IDs in PHP can be used to store unique flat data files being updated on the server without risk of overwriting other participants who may be participating concurrently. This is a very minor critique, but I felt the description of the difficulties in doing online studies with databases was a bit too dire at the end of p. 4.

Unique filename is ok using session variables however, in a technical sense, this is exactly what a database is designed to accomplish for you.

4) Clarification

I still felt unclear whether I design the data-storing facility on my own web server using psiTurk, or whether the cloud solution they offer can store data for me. I believe it is the former, but it would help to add just a bit more explicitness about this. Maybe I missed it.

This is a helpful question. There are many notions of "the cloud." Let's specifically use the psiturk.org services as our definition of the cloud. Currently, psiturk.org does not store (or really see) any of your data. It is up to the user of psiturk to set up a database using either a local text file (e.g., SQLite) or a online database like MySQL. If using MySQL, psiTurk will help you create a MySQL database on the Amazon "cloud" but this is separate from the psiturk.org services and would be tied exclusively to your personal account credentials on Amazon. We added clarification that psiturk.org never sees any of your task data explicitly.

5) General comment

One useful section could be a description of the types of studies that would be possible. There is some of this in here, of course, but more reflection on the particular directions of use (RT tasks? Categorization tasks? Decision making tasks? Etc.) would be great, even if just a short section that lays out what psiTurk will make possible (in particular with the social community aspect of the tool, which is quite exciting — what is an example domain in which the methodological and code sharing will be crucial?).

In section 4.0.3 Programming and experiment we now provide a overview of some of the experiments already listed in the exchange. We also describe how basically anything that can be programmed in the browser can be adapted for use with psiTurk.

Reviewer 3

I am glad and very thankful you sent me this before letting it go in press.

Great initiative, but I wish the authors would read and built more on what has long developed (at SCiP and in BRM)! Since the mid 90s (e.g. SCiP '96 in Chicago) several people have worked on Internet-based experimenting and other online data collection methods. Two decades of development of methods, concepts, tools for Internet-based experiments - several Advanced Training Institutes by the NSF and APA and a substantial literature - and curiously now we are seeing many initiatives like psiTurk that are trying to reinvent the wheel.
Clearly, the authors are not grounded in Internet-based research, none of them has been one of the long-time pioneers and developers of Internet-based experimenting. That may be the main reason why they are missing out on many issues and concepts that have been learned and published. They need to include those in the tool!
Please see Michael Birnbaum's 2004 Annual Review article, for example. Or my 2002 Standards for Internet-based experimenting article. Most user desires have remained unchanged from Musch and Reips' (2000) results from a survey among early web experimenters. There is also a wealth of articles in BRM and elsewhere by Birnbaum, Krantz, Göritz, Schmidt, myself and others that have not entered the authors' view. At the beginning of section 2, I was in awe when reading "It is easy to see why online experiments are an attractive option for behavioral scientists." - that is akin to saying "It is easy to see why computers could be useful in research" ;-) Really, at this point they need to go deeper and build in the depths we have long reached.

So my first feedback to the authors is: show scholarship, study the literature, foremost in the journal you have been invited to, and and connect with those who have spent their career developing the field!

We thank this reviewer for these pointers into the existing literature. To be clear, this is an invited article which was to design to introduce users to the psiTurk system. The system has some advantages over existing systems which is why it has some support among the open-source community. We are just describing the work on this project and not saying it is the best or only way to conduct online research studies.

Survey
The authors conducted a survey with ca. 200 respondents and provide the data. However, in line with Internet-based research philosophy and good practice they should provide the survey also. That way we can see whether the items were closed or open, for example. I suspect closed items

were used, because many desirable features do not appear in Fig. 2. They need to add details about the method and procedure, e.g. how were the participants recruited?
Asking an item that reads "Automatically fill in conditions randomly and evenly" reveals limited knowledge of methodology.

So my second feedback to the authors is: bring in line the open source philosophy /sharing idea and what you are doing in this ms. Provide an URL to the survey and make sure the methods and materials are available for inspection and re-use.

We have provided a link to the survey data on my lab website: http://gureckislab.org/data/online-data-survey.xlsx.

Nowhere in the manuscript I find attempts to integrate concepts and methodologies (not technologies) that were devleoped for Internet-based research. To contrast one of several tools who have been developed for experimentation online: WEXTOR.org (Reips & Neuhaus, 2002, BRM). It is platform-independent, now has almost 4000 users, and since 2000 we have continually built new features based on methodological findings into the tool, e.g.
non-obvious naming of files and conditions
simplified dropout analysis (vi_x variable)
Averts erroneous submissions (Meta tags)
Seriousness Check Technique
High Hurdle Technique
Warm-up Technique
No confounding with number and naming of pages
Downloading of data in 'each-participant-a-line-each-variable-a-column' format from an experiment
seeing a participant's path through an experiment
access to participant pool
double measurement of response times
calculated server-side response times
techniques to identify potential multiple submissions
For your reference, I'll attach the slides of my SCiP 2008 presentation on WEXTOR.

Thanks! We had never encountered WEXTOR but it seems like an interesting project. We added a new section titled “Related projects” and mention WEXTOR among others.

So my third feedback to the authors is: Include methodological concepts with the software that have been developed and validated for Internet-based experimenting. Some issues can not be fixed. If you have not fully grasped all essentials of experiment methodology and created a software that ignores important concepts (e.g. true randomization), then nothing can be done apart from a full redesign.

We are not sure what methodological concepts this reviewer is referring to. However, psiTurk is a fully open source and extensible system. People add new features daily or weekly which increase the methodological sophistication of the system. Thus, it already helps with a number of important challenges but will continue to improve (at least that is out hope!).

The fourth point I will not elaborate fully here (much more could be written), it is re technology and tech philosophy issues. Experiments created in HTML and HTML5 are also and much more directly shareable - Tim Berners-Lee advocates and defends the Open Web against more closed-in technologies and software. psiturk, despite the authors pledging to the opposite, moves away from the Open Web by introducing several components that add layers and complications to the more basic web technologies, even if they are not proprietary as some other.

This is incorrect. As mentioned on page 5, programming an actual experiment to use psiTurk can be accomplished using HTML and HTML5. The entire stack of psiTurk is open source and build on open-source Python packages like Flask and Boto. In Section 6.1 we are a little more clear about this.

The online Experiment Exchange, which catalogues existing psiTurk experiments - while seemingly a good idea on the one hand, on the other hand only becomes necessary when moving away from general web technologies (experiments in web pages written in HTML, HTML5, and Javascript) to a more closed system like psiTurk. Btw., there is a literature (again, some of it in BRM) that shows that some technologies, like Javascript, Java, Flash, interfere with study methodology and - to a degree - ethics.

psiTurk doesn't demand that the programmer use any particular system to develop the actual interface of the experiment. HTML5 and Javascript are what we use in the example code, and most of the experiment in the exchange currently use HTML/Javascript. However, it is incorrect to say that psiTurk is not consistent with open-standards. We made that very clear in the original draft of the paper but have made additional efforts on this front to avoid this misconception.

Not knowing apparently what they are getting themselves and their users into, the authors write "For example, through psiTurk, is it possible to exclude participation by workers using certain devices (such as phones or tablets) or browsers which are known to have compatibility issues with the experimenters design (for example, Firefox or Internet Explorer)." This means running into technology-sampling issues, i.e. biased samples, see Buchanan & Reips (2001) for sampling by technology effects: Mac users being more "Open to Experience" (different personality) than PC users and Javascript users having lower education than those not using Javascript.
psiTurk should not have compatibility issues with Firefox. Please. And it doesn't do Windows, which may hurt some.

psiTurk doesn't have particular compatibility issues with any browser since it provides very little in the way of web programming tools. However, the ability to exclude mobile devices is a plus since may researchers do not have to skill to create a fully "responsive" website which adapts to multiple screen sizes and input devices. We just provide the tools to block based on UserAgent. It is up to the researcher to report how this restriction was imposed in their study.

Fifth and sixth points may be usability ("UNIX-based command line interface") and dependency on a proprietary tool, Amazon's MTurk, with all methodological, ethical and legal consequences. Section "6.2 AWS credentials" reveals the system is not universal.

psiTurk is not "dependent" on AMT. In fact we use it frequently now to run laboratory studies. However, it does provide many features which help interface with AMT. That is in fact the goal of the project (the name psiturk is a reference to psychology + amazon mechanical turk).

The section "4.0.3 Programming an experiment" is totally opaque, it offers no idea or example of how one would do what the system is supposed to do after all. There is some redundancy in the ms.

We have attempted to clarify this section. The main point is that psiTurk doesn't help you program your experiment. It doesn't almost everything *besides* this. However, the Experiment Exchange allows people to share their experiment code and learn from one another.

Reading the sections on technology components, command lines, and installation bring up a seventh issue: too complicated. I doubt the average psychology researcher will master setting up the system.

Complexity in this context is a very subjective evaluation. At this point there is not data comparing the usability of psiTurk compared to other platforms. However, there are many people using psiTurk and they seemed to have benefited from the documentation.

I have not really looked much at minor issues in the ms, but off hand I spotted a few: "databases are necessary for online studies" - not true. An alternative are log files. The argument following the statement is also not true, because web servers can easily store some information in memory before it is written to a log file.

We are not sure what the reviewer is referring to with “log files.” It is not possible to create a log file on the computer of a person accessing your site using a web browser.

The part on p.5 re web server security is blurring the real issues and is in a way contradictory - either the psiTurk team makes sure to store data on a secure dedicated server (possibly within a jurisdiction that guarantees confidentiality under all circumstances, e.g. Canada, Sweden, Iceland) and under their control OR they allow and admit that data are confidential only to a limited degree, including the options they describe (any laptop, experimenter's server, cloud services). The Secure Ad Server is a nice feature, however, and increases security a bit.

This section is not contradictory at all. psiTurk.org never sees the data from the experiment. It is up to the psiturk experimenter to create and manage their own database solution and to ensure it is password protected. The data flows directly from the worker’s browser to the computer of the psiTurk experiment and is not visible to anyone else.