Open access • Journal Article • DOI:10.1145/1531793.1531815

Conference reviewing considered harmful — Source link

Thomas Anderson

Institutions: University of Washington

Published on: 21 Apr 2009 - Operating Systems Review (ACM)

Topics: Game theory, Zipf's law and User assistance

Conference Reviewing Considered Harmful

Thomas Anderson
Department of Computer Science & Engineering
University of Washington

ABSTRACT

This paper develops a model of computer systems research to help prospective authors understand the often obscure workings of conference program committees. We present data to show that the variability between reviewers is often the dominant factor as to whether a paper is accepted. We argue that paper merit is likely to be zipf distributed, making it inherently difficult for program committees to distinguish between most papers. We use game theory to show that with noisy reviews and zipf merit, authors have an incentive to submit papers too early and too often. These factors make conference reviewing, and systems research as a whole, less efficient and less effective. We describe some recent changes in conference design to address these issues, and we suggest some further potential improvements.

1. INTRODUCTION

Peer to peer systems have become a popular area of research over the past few years, reflecting the potential of these systems to provide scalable performance and a high degree of robustness for a variety of applications [5, 16, 22, 23, 24]. This line of research has resulted in substantial progress in understanding system behavior. For example, workloads, churn, and available resources are all heavy tailed, and this is fundamental to understanding aggregate system behavior in practice [6, 19]. Modeling peers as rational, sometimes altruistic, and occasionally byzantine agents [1] is essential to building systems that are both more robust and more efficient [18, 16]. And randomness is widely used in peer-to-peer systems to improve robustness [22, 6].

In this paper, we turn our attention to another peer to peer system that has received less attention from the systems research community: the systems research community itself. Our approach is somewhat tongue in cheek, but we observe many similarities, at least on the surface, between peer to peer systems and the systems research community. For one, they both lack central control! Progress occurs through the mostly independent actions of individual researchers, interacting primarily through the conference publication system.[1] Citations, and in all likelihood research reputations as well, are heavy tailed [20]. As any program committee knows all too well, authors are often rational, sometimes altruistic, and occasionally byzantine [10]. And while randomness in conference reviewing is undesirable, some have suggested that it may dominate many decisions in practice [11].

We use concepts from peer to peer systems to develop a model of computer systems research conferences. In our experience, many students and even faculty find decisions made by conference program committees to be, well, inscrutable [21]. Speaking as someone who has both authored many papers and served on many program committees, the feeling is mutual: authors often think reviewers are random or biased; reviewers often worry authors are intentionally gaming the system.

Both are right. Our thesis is that conference reviewing, as it is currently practiced today, is harmful in two ways. Conference program committees spend an enormous amount of time on what ends up for many papers being close to a random throw of the dice. Worse, conference reviewing encourages misdirected effort by the research community that slows down research progress. By illuminating these issues, we hope to blunt their impact. We also make some suggestions to better align author and conference incentives. In devising solutions, however, we urge caution: seemingly intuitive changes to regulatory mechanisms often yield the opposite of the intended effect. We give an example of one such pitfall below.

Our model has three parts taken directly from the peer to peer literature: randomness, heavy tailed distributions, and incentives. We discuss these in turn, concluding with a discussion of possible remedies. Since each of the elements of our model has been observed before with respect to research publications, we focus most of our discussion on the interplay between these elements.

2. RANDOMNESS

The task facing a technical conference program committees is easier said than done: under tight time constraints, select a small number of meritorious papers from among a much larger number of submissions. Authors would like a predictable and correct outcome, and they become legitimately upset when their papers are declined while "obviously" worse papers are accepted. While one might ascribe author complaints to the Lake Wobegon effect (everyone believes their own paper is above average) [10], authors with multiple submitted papers have a unique perspective: did the PC ranking match their own? Often the answer is no.

How can this be? In computer systems research, individ-

[1] In computer systems research, peer-reviewed conferences, rather than journals, are the primary way that research results are disseminated.

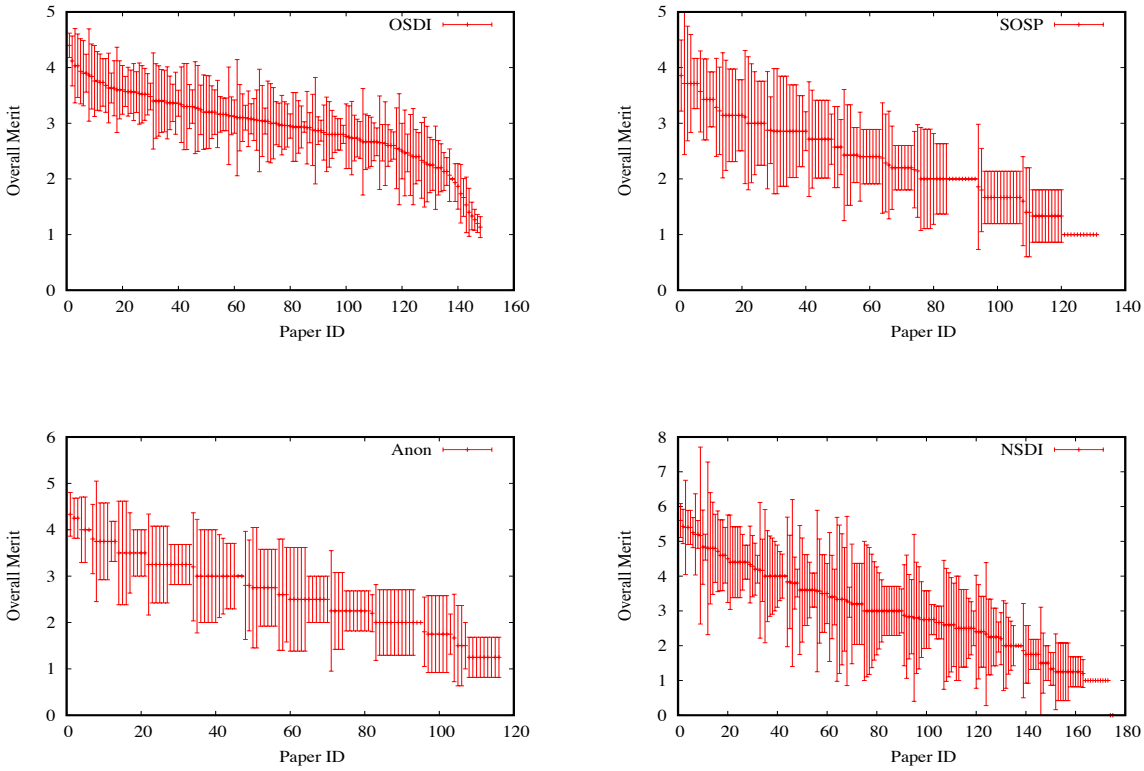

Figure 1: Mean evaluation score with standard deviation, for each paper submitted to four recent systems research conferences. Papers are sorted by mean review score.

ual reviewers differ significantly on the very fundamental issue of what is merit: how much to weight various factors such as likely future impact, importance of the topic, uniqueness and creativity of the solution, thoroughness of the evaluation, and effectiveness of the presentation [2, 21]. Some reviewers penalize good ideas that are incompletely evaluated, as a spur to encouraging authors to complete their work prior to publication; others do the opposite, as a way to foster follow-up work by others that may fill in the blanks. Some reviewers are willing to accept papers that take a technical approach that they personally disagree with, as long as the evaluation is reasonable; others believe a program committee should attempt to prevent bad ideas from being disseminated.

Even if reviewers could somehow agree on all these factors, the larger the program committee, the harder it is to apply a uniform standard to all papers under review. Systems research conferences have seen a rapid increase in the number of papers submitted. Some have suggested charging authors per submission [7] as a way of reducing the flood. However, the rate of production of scientific research papers has been doubling every fourteen years for the past several centuries [8]. The rate of systems research papers is likely to increase, no matter what mechanism is adopted. To deal with this, either the workload of a given program committee member, or the size of the program committee, or the number of conferences, must continue to increase. Or all three.

Anyone who has served on a top tier program committee understands the result: altogether too much randomness in the outcome, despite a herculean effort to provide every paper with a significant number of detailed reviews. Figure 1 shows the mean and standard deviation (square root of the variance) among review scores for papers submitted to four recent first-tier systems conferences: OSDI 2006, SOSP 2007, NSDI 2008, and another that shall remain anonymous. Papers are ranked by average review score, with error bars for the standard deviation among scores for each paper. Each of the conferences accepted between 25-30 papers. In most cases, reviewers were permitted to revise their scores after reading other reviews, but few chose to do so. Hence Figure 1 largely reflects the underlying variance in reviewer opinion, rather than the consensus that emerged from the program committee meeting. Figure 2 reports the number of reviews for each paper, ranked in the same order as Figure 1; if the review scores for a particular paper are iid around a common mean, one would need to *quadruple* the number of reviews to halve the standard deviation, not a particularly welcome scenario for already overworked program commit-

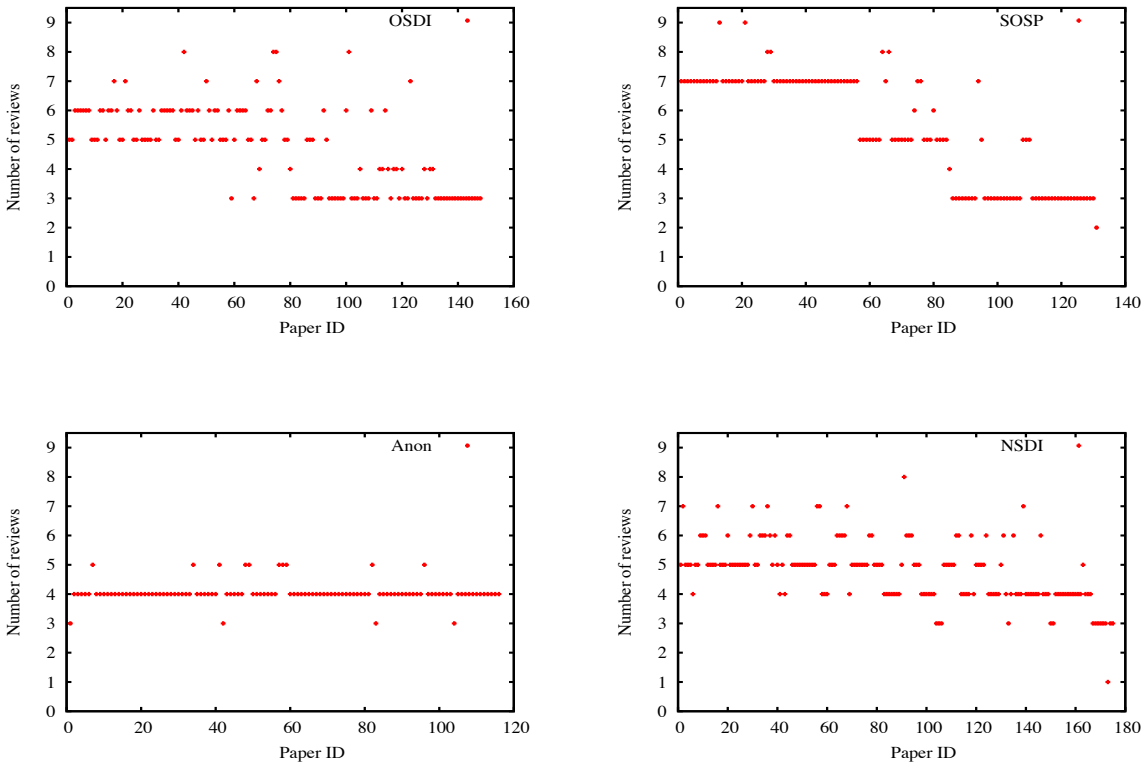

Figure 2: Number of paper reviews, for each paper submitted four recent systems research conferences. Papers are sorted by mean score, as in Figure 1.

tees.

As the figure shows, the variance in reviewer scores is far larger than the difference in the mean score, for a broad range of papers around the boundary between accept and reject. All four conferences held a conventional program committee meeting; papers were accepted after a discussion, not solely based on the mean score.

While it might seem intuitive that the program committee discussion adds value, some caution is merited. Several times over the past few years, a comparison between the programs chosen by a shadow PC and the real PC have shown a remarkable lack of congruence. One could explain this as due to the relative lack of experience of the shadow PC, but an equally plausible explanation is simply random chance in the assignment of reviewers [11]. In a recent case, a paper rejected by one systems conference was an award paper at the next, with minimal changes in between. The subjective notion that program committee discussions add value can be at least partly explained as an artifact of human cognition; when forced to make a choice between nearly equivalent options, the human brain will make up reasons post hoc for why the differences were in fact significant [12].

If we cannot eliminate randomness in the paper selection process, we should at least actively manage it. For SIGCOMM 2006, Nick McKeown and the author developed a review process that aimed at cost-effectively using scarce reviewing cycles. Papers at either end of the quality spectrum were reviewed less often than papers at the margin. Papers with high variance in review score were automatically given additional reviews, while those with no variance were not. One particularly controversial paper received nine reviews (and a half hour discussion at the PC meeting) before being accepted, others were rejected after two reviews, or accepted after four. Further, by seeking reviews whenever there was variance, we were able to assign reviewers as reviews rolled in, rather than having to hold up each phase of assignments for the inevitable PC laggard. In a committee with fifty members, someone will always be late.

Before reviewing started, we engaged the program committee in a collective discussion as to how to weight the various factors of merit discussed above, and we carefully chose the questions on the review form to reinforce that social consensus. Cognitive experiments suggest that value judgments can be significantly influenced by subtle reminders [12].

To manage workload and improve confidence intervals, we had a relatively large program committee split between "heavy" and "light". All program committee members were assigned approximately 15 papers to review, and essentially

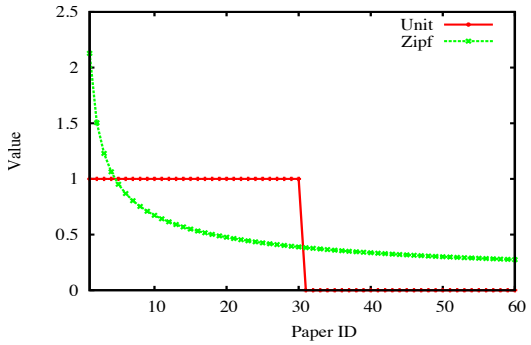

Figure 3: Two alternate models of the distribution of research merit. Both curves have the same aggregate value.

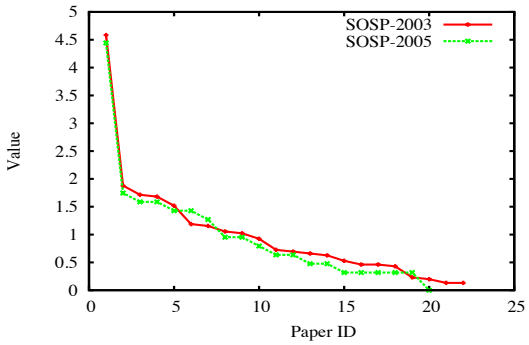

Figure 4: Normalized citation count for two recent SOSP conferences.

asked to bin their set of papers into strong accept, marginal accept, marginal reject, and reject, proportionately to those numbers in the overall program. (In this fashion, we hoped to force reviewers to make a judgment call – would they include this paper in the program? Otherwise a large number of papers would end up in the marginal camp, and no information would be conveyed by the review score.) Initial paper assignments were done randomly among program committee members, among those qualified in the paper topic area, to further improve the confidence intervals. We used external reviewers only to provide missing expertise.

Based on score, variance, and an email consensus, we preaccepted nearly half the papers at the conference prior to the program committee meeting, so that we could focus the in-person discussion on precisely those papers for which the answer was least clear. The light program committee was not asked to travel; the heavy program committee met in person to consider the papers at the margin. To improve consistency, each paper under discussion was read by at least a quarter of the heavy PC, hence the term, "heavy PC".

Readers are invited to judge for themselves whether the quality of the program differed significantly from other iterations of the same conference. What we did find, however, was somewhat surprising: we nearly drove the heavy PC insane. The process efficiently identified the set of papers which could be easily decided, leaving the heavy PC to grapple with a large number of incomparables. Is one paper's incomplete evaluation more or less important than another's narrow applicability? The difference between an accepted and a rejected paper is hugely important to the authors, and yet, in the end, there was little consistent basis to decide between papers, beyond the few clear accepts. Explaining why our PC found it so difficult to rank papers is the topic of the next section.

3. ZIPF

In this section, we consider how merit is distributed among the papers submitted to a conference. To make the discussion concrete, Figure 3 draws two plausible alternate models. In the first model, we assume a conference in which thirty papers are accepted, all accepted papers have the same value, and no rejected paper has any value. In the second model, we assume paper value is distributed according to a zipf curve, $x^{-1/2}$ for the top sixty submitted papers, with the other papers having zero value. The curves are scaled to have the same total value.

There is a widespread recognition that simple paper counting is an invalid way to determine the impact of a researcher's career. After all, there are so many publication venues, that surely some venue will publish virtually any valid paper. The length of a CV does not indicate much of anything, beyond effort. Nevertheless, with respect to any single conference or journal, especially well-known ones, paper counting is a widespread practice. This is for good reason: with no additional information about the papers other than their titles, all the papers accepted at a given conference are equivalent, and all rejected papers are unknown. Program committees encourage paper counting by providing no ranking information among accepted papers, except in some cases to identify a small number of award papers. By default, then, this leads to the valuation function as drawn in Figure 3: a step function where papers that are accepted are all equally valued, and papers that are rejected are, for all practical purposes, worthless.

Obviously, a step function is a poor approximation of the underlying merit of a set of submitted papers, and we will argue later that using a step function, even in part or by default, incents researchers inappropriately. In earlier work, we showed that the step function reward curve used in BitTorrent makes it particularly vulnerable to strategic client behavior [18].

But first, how is merit distributed? A full characterization of research value is beyond the scope of this paper, but we believe a zipf distribution may be a reasonable approximation. In a zipf distribution, if x is the rank of an item, $f(x) = x^{-\alpha}$, where α can vary between 0 and 1. Zipf distri-

butions have been found to model many apparently different structures, such as the frequency of words in books, the size of cities, the popularity of movies and web pages, and so forth.

Zipf distributions model most social processes, so it should not be surprising that they are also helpful in explaining computer systems research. Specifically, a zipf distribution captures the intuitive notion that most papers submitted to a particular conference have something useful to say. Papers at the very top of the accepted list are often quite a bit better than the others (there's even a phrase for this: "clear accept"). But there are not many clear accepts, and for the remainder, there is precious little difference between accept and reject. As shown in Figure 3, with $\alpha = 1/2$ and thirty accepted papers, assuming the program committee was perfect in its judgment, the difference between the worst accepted paper and the best rejected paper is approximately 1%. There is no reviewing system that we know of that can reliably distinguish that level of difference (or anywhere close to it), and so in practice, given the randomness discussed in the previous section, the best rejected paper may be quite a bit better than the worst accepted one.

We note that citations in scientific research in general [20] and computer science in particular [3, 17], are distributed according to a zipf curve, with $\alpha = 1/2$, at least for the top 10-20% of the total universe of published research papers. (The rest are write once, read none.) Since most readers will find it implausible that systems papers are zipf, we illustrate it using two recent SOSP conferences. In Figure 4, we plot the normalized citation count (that is, the citations to a specific paper, normalized by the average citations among all papers appearing in the conference) for all papers published at SOSP 2003 and SOSP 2005. Note that the program committees for those conferences most likely (and quite rightly) did not select papers solely for their citability. It would of course be interesting to compare PC ratings with later citation counts, but we lack the data to determine this. Given the randomness of PC reviews, we suspect any such relationship to be weak. Nevertheless, on this one metric, it is interesting that there is a 5-10x difference between papers appearing at the same conference.

In our view, citation counts do not represent all, or even most, of the intrinsic value of research papers. Rather our point is simply that a zipf curve is a more realistic model of research value than a unit model that considers only the fact of publication at a specific venue. Citation counts typically mix all sources of citations together, regardless of the merit of the conference or the depth of the contribution implicit in the citation. More fundamentally, citation counts favor the early over the definitive. To take an extreme example, the most referenced computer systems research paper ever published (according to citeseer [14]) is the initial TCP congestion control paper [13]. While that paper is influential, it would be hard to argue that it is more meritorious than every other systems paper that has ever been published. Rather, the TCP paper, like a large number of widely cited papers, was (i) early, (ii) left ample room for others to innovate, and (iii) was in a research area that had a low barrier to entry for other researchers (in this case, because of the widely used simulation package, ns2). Only some of those three characteristics could be considered inherently valuable.

One implication of a zipf distribution of merit for conference submissions is that, for a popular conference, the aggregate value of the rejected papers may be comparable to or even larger than the aggregate value of the accepted ones. Zipf is a heavy-tailed distribution, which means there is significant merit under the tail of the curve. Of course, the result is somewhat different if we consider value per square inch of paper!

Noise in the evaluation system and zipf distribution of paper merit fosters grumpiness among both authors and program committees [15]. If authors systematically overestimate the quality of their own work [10], then any paper rejected near the threshold is likely to appear (to the author) to be better than a large percentage of the actual conference program, implying (to the author) that the program committee was incompetent or venal. When a program committee member's paper is rejected, the dynamic becomes self-sustaining: the accept threshold must be higher than the (self-perceived) merit of their own paper, encouraging them to advocate rejecting even more papers. Because of memory bias [12], reviewers are more likely to remember the good papers than the bad papers from earlier conferences, so that reviewers often believe an even higher threshold has been applied in the past. It is a wonder that program committees accept any papers at all. Fortunately, that would be too embarrassing.

All this might give support to advocates of conferences with parallel tracks. After all, why not simply accept all valid papers, and run as many parallel tracks as necessary? All things being equal, this would maximize the information content of the conference, compared to one that picked an arbitrary threshold. However, from the audience perspective, the information content of a multi-track conference is strictly less than a single track one. The typical conference attendee is faced with the conundrum that the best papers at the conference are spread thinly across all sessions. Multitrack conferences also run afoul of incentives, our next topic.

4. INCENTIVES

We next turn our attention to the role of incentives in conference design, particularly the interplay between incentives, randomized selection, and heavy-tailed merit. Clearly, a full investigation of researcher incentives is beyond the scope of this paper. It seems likely that no two researchers share the same set of motivations, other than that it is unlikely to be monetary reward! For the purposes of discussion, we assume researchers are motivated in part by research recognition, recognition is given in part by publication at prestigious venues, and recognition is unit-value based on the venue (the

average of the true merit of all papers that appear there). Of course, this model is unrealistic in that most researchers (we hope!) are not primarily motivated by the mere fact of publication. We further assume merit is heavy-tailed among the universe of publications in a particular research area, and authors are aware of the relative merit of their own work – another dubious assumption!

Under these assumptions, authors of better papers essentially subsidize the research reputation of the conference and indirectly, the inherited reputation of marginal papers. From this, it is easy to see why it is rare to find high prestige multi-track conferences. As with any cross-subsidy, there is an incentive for the subsidizers to avoid the tax. Authors of better papers have an incentive to send their papers to a more selective single-track conference, and if one didn't exist, they would have an incentive to band together to create one. (Equivalently, conference organizers have an incentive to attract high merit papers by being more selective.) Moreover, once such a single-track conference was successfully created, authors of other papers would be incented to send their papers to the single-track conference, as they would benefit by association. Under the unit value model, it is better to be a worse paper at a good conference than a good paper at a worse conference.

By contrast, a stable situation is the one we often see in practice. First, there is often a high-prestige single track conference, with award papers, to capture and reward the top end of the zipf distribution. Since that inherently leaves a large number of unpublished papers of significant value (nearly equal to the intrinsic value of the median paper at the high prestige conference!), those authors would be incented to send their papers to a different venue. The second conference is more usefully multi-track, as there is less difference between successive papers as we go down the distribution. In a zipf distribution with $\alpha = 1/2$, the difference between paper 30 and paper 100 is the same as the difference between paper 8 and paper 30.

We conclude with an observation. It should be the goal of conference design to encourage authors to complete their work to the point of diminishing returns. Great ideas should be thoroughly fleshed out, while mediocre ones should be quickly documented, allowing the authors to move on to greener pastures. We might come close to this if reward followed merit. Zipf reward would provide increasing returns for increasing effort; it is much better to be the most influential paper at a conference than the tenth best. However, under the current system of multiple conferences per year per area, unit value reward for a specific conference, and noise in the evaluation system, authors have an incentive to do considerably less than this ideal.

Consider the marginal incentive for an author of a research paper. By marginal incentive, we mean the incremental benefit to the author of putting additional effort into improving a particular paper. After all, the author does have a choice: put more effort into an existing project, or start a new one. Some authors make a virtue of this, by going after "low hanging fruit." (To push the analogy, cognitive experiments indicate that the higher the fruit, the tastier it will be [12]. So perhaps we should strive to avoid low hanging fruit?) Suppose we posit a noiseless system with a single conference, unit reward for getting papers into the conference, and perfect information, e.g., knowledge by the author of the threshold to be applied by the program committee. The author's marginal incentive in such a system would be an impulse function: do no work unless the idea is above threshold, and then do only enough work to push it above the threshold for publication. Some might say that SOSP prior to 1990 was such a system. The conference met once every two years (reducing the opportunity for retries), there were few alternate venues of comparable merit, the threshold was high (simulation studies need not apply), and much of the community could guess whether a particular paper idea had a chance. The result was a relatively small number of high quality submissions.

Of course, SOSP was not a perfect system. As we noted earlier, a substantial fraction of the aggregate merit is likely to be in the tail of the curve, missed by a highly selective process. In at least one case, a paper rejected at one SOSP was followed up by five papers by other research groups at the next SOSP. Memory bias led to a gradual raising of the threshold at the SOSP, until the program barely filled two days. Eventually, the threshold was lowered by fiat, but that has its own problems. With unit value return, authors have an incentive to only do the minimal effort necessary to pass the threshold, leading to more mediocre papers being submitted and published.

Uncertainty changes the results a bit. Reviewing noise and an uncertain threshold means that an author cannot perfectly predict whether a particular paper will be accepted. Thus, the marginal incentive for an author, considering a single conference at a time, is gaussian: increasing as it approaches the likely threshold, and then falling off as the paper becomes increasingly likely to be accepted. Oddly, the *more* randomness in the evaluation and the less predictable the outcome, the more incentive authors have to work harder. Slot machines work on a similar principle: no one would ever play a slot machine that simply returned the expected value every time. In other words, efforts to reduce randomness in conference reviewing may in fact be counterproductive to research progress.

Repeated trials also changes the equation. In many areas of computer systems research, we have prestigious conferences every six months. This is a direct consequence of increasing competition – as we observed, the universe of papers is increasing exponentially, and authors of good papers at weak conferences have an incentive to try to create higher prestige venues for their papers. The multiplicity and frequency of venues provides authors an alternate strategy to compensate for unpredictable program committees: submit papers initially with the minimal amount of work needed to be competitive. If accepted, move on. If rejected, add some

work and repeat.

Because of randomness and repeated tries, a persistent and strategic author will be able to get papers accepted with much less work than an author with only one shot. This can result in a race to the bottom. The more competition there is for publication, the more conferences we have, the lower everyone's threshold becomes, and the more incentive authors have to stop when their papers pass that minimal threshold.

We are of course not recommending this course of action – an author focused on long-term impact is probably better served by ignoring the signal of publication counts, even at top tier conferences, as being too noisy and too uniform. Rather, if long-term merit is zipf distributed, then a better strategy would be to "put more wood behind fewer arrows". The main point of this paper is that conference reviewing sets up a system of short term incentives, that if followed, lead to sub-optimal behavior by researchers. To encourage a longer term view, it would be ineffective for program committees to reject papers that are great ideas, but incompletely executed: in that case, strategic authors would have an incentive to work only on those research ideas that are obviously mediocre from the start.

5. IMPROVEMENTS

Most of these counterproductive incentives would disappear if we were able to set rewards to model the underlying long-term value of the work. Provided noise is a second order effect, authors would be incented to complete the work to the point of diminishing returns. Progress in the field would move fastest if authors were incented to hold back papers until they were ready, or equally, to publish a short form of the research idea early followed by a later, more thorough evaluation.

Several recent papers suggest various ways to improve on conference reviewing. One path might be to revitalize journals [4]. If the systems research community shifted to value journal over conference publications, journal reviewers and editors could become the arbiter of merit instead of time-constrained and overworked program committees. However, journals are likely to also have noisy reviews and zipf merit, and thus are likely to suffer from the same counterproductive incentives as conferences. Computer systems research is not alone in this. Ellison presents a model for how journal publication in economics is being distorted by incentives [10], leading to overpolished and uncreative work. More crucially, it is hard to see how we get from here to there. Unless authors send their best work to journals, journals will remain thinly read and less selective than top tier conferences, incenting authors to skip the additional work necessary to archive their work, leading journals to remain thinly read.

Crowcroft et al. suggest charging authors, in a virtual currency, for the privilege of submitting to conferences, as a way of discouraging the practice of submitting too early and too often [7]. Authors submitting good papers would be refunded the submission cost, while authors submitting poor papers would not. New authors would be granted tokens at a slow default rate, and all authors can collect more tokens by providing well-regarded reviews of other papers, addressing the workload problem for program committees. While this proposal is intriguing, the experience with peer to peer systems is that virtual currency systems are easier to design than to get to work in practice; BitTorrent is resilient and widely used precisely because it requires no centralized currency or trust [19].

Douceur, in a companion paper in this issue, advocates asking reviewers to rank papers, instead of to rate them [9]. A global ranking is computed from individual pairwise rankings of papers, as a starting point for the program committee discussion. Using ranking instead of ratings should reduce noise somewhat, because the ranking provides strictly more information. For this reason we are planning to use ranking in the SOSP 2009 program committee. But ranking is unlikely to be a silver bullet: different reviewers will still differ as to the definition of merit, and if merit is zipf-distributed, the absolute value of the difference in merit between adjacent papers is likely to be small.

What is to be done? The conference organizer is faced with a dilemma. Raise the threshold by accepting fewer papers, and retard progress by delaying good work in the heavy tail. Or lower the threshold by accepting more papers, and establish a new lower standard for authors to iarget. The main goal of this paper is to simply raise this issue in public, in the hope that authors will decide on their own to take a longer term view.

Beyond that, our recommendation is to focus on increasing transparency of the program commmittee process. Incentive systems work best when all participants have the same information; they are most vulnerable to strategic manipulation when information is hidden or available to only certain participants. Public reviews, instituted by Nick McKeown and the author for HotNets 2004 and SIGCOMM 2006, are a step towards transparency, but they do not go far enough. With public reviews, the program committee publishes the rationale for why they accepted each paper, along with an assessment of how the paper fits into the broader research context and the paper's strengths and weaknesses. Many students have reported that the public reviews were the most valuable aspect of SIGCOMM 2006. However, public reviews impose a high overhead on conference program committee, as the public review is permanent and therefore must be carefully authored. Thus, it is not clear if writing public reviews is a sustainable practice.

We make four suggestions to further improve transparency. None of these ideas is particularly new; we gather them here because they are motivated by the model of conference reviewing we have developed in this paper.

- No review without publication on the web. Submissions to systems conferences are, by tradition, reviewed confidentially. Accepted papers are revised based on the reviews, but rejected papers may never appear. This has

two negative consequences. As we noted, much of the aggregate merit in a set of submitted papers is in the tail of the curve; if these papers are delayed, the research community is worse off for it. Instead, we should encourage work to appear as early as possible, and to be revised and improved over time. Second, authors today often submit papers before they are ready [7], simply because there is a chance that the program committee will take the paper anyway. In our view, if authors are unwilling to place their names on their work in public, then that is a signal that their paper is unlikely to be ready for anyone, including reviewers, to read it.

- Encourage community review of web publications [7]. Simply posting every paper to the web is insufficient of course, because the number of papers is scaling exponentially. Readers need some way to navigate through the thousands of systems research papers written each year, and the most scalable way to do that is for readers themselves to provide recommendations as to what is worth reading. Eventually, web publication may reduce the demand for conference publication. Ellison, for example, reports that top tier economics faculty no longer submit their papers to journals, because people read their work anyway [10]. If everyone already knows of your work, there is little incentive or benefit from formal publication. This would leave research conferences to focus on what they are uniquely qualified to do: to find and publicize the best previously unknown research.
- Conferences should publish the program committee's ranking and confidence interval for each submitted paper. The rank and error bounds could be automatically computed from the rankings of individual reviewers, or it could be an explicit output of the program committee meeting [9]. Traditionally, the program committee's internal rating is kept confidential because it is nearly meaningless – as we noted earlier, the error bounds on paper ratings dwarf the differences between the mean ratings, in many cases. Nevertheless, the research community would benefit from knowing what precisely the program committee is saying (or not saying) about a paper's merit, by its accept and reject decisions. This would also reduce the incentive for authors to submit papers that narrowly beat the threshold, or conversely, are far below the threshold.
- Re-review and publically rank conference papers after an interval. A program committee's initial assessment of a research paper's merit may bear only a small relationship to its long term merit. This seems inevitable given the time constraints of the conference evaluation system. While journals might be an avenue for this long term perspective, in a fast changing field field such as computer systems research, more attention is naturally paid to the most recent results. To better align long term researcher incentives, and to encourage researchers to continue to work on promising research avenues even after the initial publication of the work, we advocate adding a step to publically re-rank the set of published papers after some period of time has elapsed. In many areas of computer systems research, "Test of Time Awards" are given to the retrospectively most important paper published at a specific conference ten years earlier. Since it is likely that the merit of research papers is still heavy tailed after the test of time, we suspect the research community would benefit from being given a ranking of papers, rather than information about only a single paper.

6. CONCLUSION

In this paper, we have developed a model of computer systems conference publication based on randomness in paper evaluation, heavy-tailed merit, and author and conference incentives. We hope this model will be helpful in explaining to prospective authors the often obscure workings of conference program committees, and in encouraging systems researchers to ignore mere publication as a figure of merit. We suggest several changes to increase the transparency of conference reviewing, with the twin goals of improving the efficiency of the reviewing process, and more importantly, of better aligning author incentives with what is needed for rapid and sustained progress for the field of computer systems research.

Acknowledgments

The author would like to thank Arvind Krishnamurthy, Eddie Kohler, Frans Kaashoek, Jeff Mogul, Jon Crowcroft, Mike Dahlin, and Derek Murray for their help in assembling the data presented in this paper. We would also like to thank Stefan Savage, Jeff Mogul, Steve Gribble, and Ed Lazowska for numerous suggestions that substantially improved the presentation.

7. REFERENCES

[1] A. S. Aiyer, L. Alvisi, A. Clement, M. Dahlin, J.-P. Martin, and C. Porth. BAR fault tolerance for cooperative services. In *SOSP*, 2005.

[2] M. Allman. Thoughts on reviewing. *SIGCOMM CCR*, 2008.

[3] Y. An, J. Janssen, and E. E. Milios. Characterizing and mining the citation graph of the computer science literature. *Knowl. Inf. Syst.*, 6(6):664–678, 2004.

[4] K. Birman and F. Schneider. Program committee overload in systems. *To appear, Communications of the ACM*, 52, 2009.

[5] S. Buchegger and J.-Y. L. Boudec. A robust reputation system for P2P and mobile ad-hoc networks. In *Second Workshop on the Economics of Peer-to-Peer Systems*, 2004.

[6] B. Cohen. Incentives build robustness in BitTorrent. In *Workshop on Economics of Peer-to-Peer Systems*, 2003.

[7] J. Crowcroft, S. Keshav, and N. McKeown. Scaling the academic publication process to internet scale. *Communications of the ACM*, 52(1), 2009.
[8] D. J. de Solla Price. Networks of scientific papers. *Science*, 149(3683):510–515, 1965.
[9] J. Douceur. Paper rating vs. paper ranking. *ACM SIGOPS Operating Systems Review*, 43(2), 2009.
[10] G. Ellison. Evolving standards for academic publishing: A q-r theory. *Journal of Political Economy*, 2002.
[11] A. Feldmann. Experiences from the SIGCOMM 2005 European shadow PC. *SIGCOMM CCR*, 35(3):97–102, 2005.
[12] C. Fine. A mind of its own: How your brain distorts and deceives. Icon Books, 2006.
[13] V. Jacobson. Congestion avoidance and control. In *SIGCOMM '88*, pages 314–329, 1988.
[14] S. Lawrence, C. L. Giles, and K. Bollacker. Digital libraries and autonomous citation indexing. *IEEE Computer*, 32(6):67–71, 1999.
[15] R. Levin and D. Redell. An evaluation of the ninth SOSP submissions; or how (and how not) to write a good systems paper. *ACM SIGOPS Operating Systems Review*, 17(3):35–40, 1983.
[16] R. Mahajan, D. Wetherall, and T. Anderson. Mutually controlled routing with independent ISPs. In *NSDI*, 2007.
[17] V. Petricek, I. J. Cox, H. Han, I. Councill, and C. L. Giles. A comparison of on-line computer science citation databases. In *Ninth European Conference on Research and Advanced Technology for Digital Libraries (ECDL 2005)*, 2005.
[18] M. Piatek, T. Isdal, A. Krishnamurthy, and T. Anderson. Do incentives build robustness in BitTorrent? In *NSDI*, 2007.
[19] M. Piatek, T. Isdal, A. Krishnamurthy, and T. Anderson. One hop reputations for peer to peer file sharing workloads. In *NSDI*, 2008.
[20] S. Redner. How popular is your paper? an empirical study of the citation distribution. *European Physical Journal B*, 4(2):131–134, 1998.
[21] S. Shenker, J. Kurose, and T. Anderson. Improving SIGCOMM: A few straw proposals. In *www.sigcomm.org/admin/July2001RepFinal.pdf*.
[22] I. Stoica, R. Morris, D. Karger, F. Kaashoek, and H. Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. In *SIGCOMM*, 2001.
[23] V. Vishnumurthy, S. Chandrakumar, and E. Sirer. KARMA: A secure economic framework for peer-to-peer resource sharing. In *Workshop on the Economics of Peer-to-Peer Systems*, 2003.
[24] P. Yalagandula and M. Dahlin. A scalable distributed information management system. In *SIGCOMM*, 2004.