The SIGMOD 2019 Research Track Reviewing System

Anastasia Ailamaki[†‡] Periklis Chrysogelos[†] Amol Deshpande[¶] Tim Kraska[§]

[†]EPFL, [‡]RAW Labs SA
{firstname}.{lastname}@epfl.ch

[¶]University of Maryland
amol@cs.umd.edu

[§]MIT
kraska@mit.edu

1. INTRODUCTION

While organizing the submission evaluation process for the SIGMOD 2019 research track, we aim at maximizing the value of the reviews while minimizing the probability of misunderstandings due to factual errors, thereby valorizing impactful ideas. The objective is an educating and rewarding experience for both the authors and the reviewers.

The actionable goals are:

1. Maximize review depth and breadth. For depth, optimizing the assignment of papers to reviewers is of key importance; "low confidence" reviews should be few to none, in order for reviewers to provide extensive and useful comments to the authors. To cover the breadth and to address controversial issues, recruit as many reviewers as needed to converge to a unanimous set of comments.

2. Ensure that all submissions are treated equally fairly by experts in the respective domains.

3. Obtain as much input from the authors as possible during the process. Enabling author feedback is the key step in the process.

4. Allow re-evaluation of papers with non-critical flaws through revisions.

The better part of the infrastructure is devoted to optimizing reviewer assignments and to orchestrating the first set of reviews and discussions, both of which are critical parts in the entire process. The rest of this paper describes the process we set forth and our experiences from running it.

2. REVIEWER ASSIGNMENT

We address two issues in the reviewing process.

PC member load and responsibilities. Reviewer load and responsibilities must be clear and predictable. Some research areas are at times more popular than others; when selecting reviewers, we make sure that the PC composition reflects topic popularity, albeit not always perfectly. Trying to achieve uniform load across all reviewers therefore inevitably results in non-expert reviews. In addition, after reviews are submitted, a discussion starts which ends at the very end of the process with a decision which is summarized in a meta-review and is supported by final, polished reviews. Orchestrating the discussion and ensuring final review quality is a major responsibility and a critical step toward an unanimous and educated decision. Top database conferences employ a two-level reviewer hierarchy: A large set of reviewers who provide the reviews and a smaller set of meta-reviewers who act as area chairs for a subset of papers (typically four times as many as the average reviewer load). The advantage is that the work is distributed and the quality of the final result is increased. Nevertheless, the meta-reviewers can only help resolve a disagreement if they have read the paper in question themselves, which is only partly possible and typically happens under severe time constraints (after the end of the review period).

Reviewer assignment. The typical method for reviewer assignment relies on topic relevance: reviewers of a paper with intellectual contributions in a certain topic provide useful comments if they work on the same or a related topic. Feedback from recent conferences, however, shows that the evaluation methods play a significant role in the appreciation of a paper's contributions. The evaluation of a paper's contributions is typically based on theoretical proofs or on observations from experiments with implemented systems. Experience shows that a paper is evaluated most fairly when at least a subset of its reviewers are experienced on the evaluation methods the paper uses.

The rest of this section describes how we structure the program committee work in order to resolve the aforementioned issues.

2.1 PC member load and responsibilities

The number of assigned papers may vary across program committee members – this is due to unexpectedly disproportional number of submissions in a subset of topics. This is a fact that reviewers need to be alerted about. Reviewers also need to be prepared to provide impromptu reviews in severe conflict cases.

A subset of reviewers form the **core committee** and are responsible for (a) providing reviews for their assigned papers, monitoring discussions, ensuring high review quality, and writing meta-reviews as needed. Core committee members (aka meta-reviewers) are assigned about double the number of papers that **regular committee** members are assigned; the latter are responsible for providing reviews for their assigned papers and for participating in discussions. Reviewer invitations are clear about load and responsibilities. In SIGMOD 2019, about 25% of the reviewers are core committee members.

For review assignments, we collect a number of attributes that describe reviewers and submissions to enrich the data provided by the conference management tool and provide better assignments.

2.2 Reviewer assignment preparation

The goal of assigning papers to reviewers is to maximize the average reviewer expertise per paper, and minimize the number of non-expert reviews. To maximize the level of expertise per paper we characterize papers and reviewers with respect to the style of evaluation and contributions.

Paper characterization. The SIGMOD 2019 paper submission form asks authors to characterize their paper as either **systems** or **non-systems**. The call for papers describes **systems papers** as:

- Papers that describe an entire new system, covering, e.g., the system architecture and design issues or experiences learned from building the system.
- Papers that extend an existing (open-source) system with new or more efficient functionality. Such papers may add new functionality to Spark, Hadoop, PostgreSQL, etc. The motivation may be to better support new applications.
- Papers that concern specific aspects of a system or systems. Such papers may concern storage management, query processing, indexing, transaction management, access control, authentication, etc.
- Papers that concern systems support for new hardware, e.g., multi core, SIMD, NUMA, HTM, SGX, GPU, FPGA.
- Papers that analyze system performance.

The systems/non-systems annotation helps ensure that system papers are assigned not only to reviewers who are experts on the topic of the papers but also have extensive expertise in writing and evaluating systems papers.

Reviewer categorization. Similarly to papers, each program committee member is annotated as "systems" or "non-systems" according to their own published work. As a result, each of the core and regular program committees is further divided into systems and non-systems subcommittees. (To preserve reviewer anonymity, the systems annotation is not announced on the website.)

To match reviewer and paper research areas, we use both the Conference Management Toolkit (CMT) [4] and the Toronto Paper Matching System (TPMS) [3]. Upon acceptance of the invitation to the program committee and prior to abstract submission, all reviewers mark their research areas on CMT and upload a set of their own representative publications on TPMS. After paper submission, all submissions are uploaded to TPMS that outputs baseline matching scores for each reviewer-paper combination.

Using TPMS scores as a similarity measure. TPMS analyzes the submitted papers and the reviewers' uploaded list of papers and produces a score based on their topics similarity, extracted based on word counts and LDA [2].

Preferred reviewer similarity measures. We solicit proposals for reviewers from both paper authors and core committee members as follows. At submission time, authors can optionally propose "preferred" reviewers (inside or outside the program committee) for their submissions, which we use opportunistically as an additional indicator of the quality of a reviewer-paper combination. Similarly, at the first submission cycle, we asked meta-reviewers to suggest reviewers for each of their assignments. Handling meta-reviewer input, however, was manual and incurred more overhead than its associated benefit, so we dropped this option in the second submission cycle. Nevertheless, meta-reviewer input can prove useful if the suggestions are incorporated into the process automatically.

Conflicts of interest. We augment the conflicts of interest inserted by authors and reviewers through CMT with additional ones based on the mined collaboration graphs provided by AMiner [1]. This was a useful step as none of the COI lists was a superset of the other: AMiner is not aware of ongoing collaborations and conflicts outside of what can be inferred from published work or employment status, while authors may forget to register some conflicts.

2.3 Meta-reviewer assignment

We assign meta-reviewers to submissions using an expanded variant of the integer program proposed by Taylor [7] and reused in TPMS [3]:

$$\begin{aligned}
\underset{y_{rp}}{\text{maximize}} \quad & \sum_r \sum_p \hat{s}_{rp} y_{rp} \\
\text{subject to} \quad & y_{rp} \in \{0,1\}, && \forall r,p \\
& \sum_r y_{rp} = R, && \forall p \\
& y_{rp} = 0, && \forall (r,p) \in COI \\
& \sum_p y_{rp} \leq S_{max}, && \forall r \\
& S_{min} \leq \sum_p y_{rp}, && \forall r
\end{aligned} \tag{1}$$

where, y_{rp} is constrained to 0-1, with $y_{rp} = 1$ if and only if reviewer r is assigned to paper p, R is the number of required reviews per paper and S_{min}, S_{max} is the minimum and maximum number of assignments per reviewer, $\hat{s}_{rp}$ is the gain from matching reviewer r to paper p and COI is the set of conflicts of interest. We observe that for large values of S_{max} and as there is no lower bound, two groups of reviewers are created: the "misunderstood" ones who are assigned a very small number of papers and the "rockstars" that are assigned almost S_{max} papers, with almost no one in between. To avoid bipolar groups, we augment Taylor's formulation by setting a lower bound S_{min} for the number of assignments per reviewer. For meta-reviewer assignments, r is simply restricted to the set of meta-reviewers and, as we assign only one meta-reviewer per submission, $R = 1$.

For SIGMOD 2019, we use:

$$\hat{s}_{rp} = s_{rp} w_{rp} + \mathit{refinement}_{rp} \tag{2}$$

where s_{rp} is the TPMS score, $\mathit{refinement}_{rp}$ is added to allow penalizing or embracing matches after manual inspection of the results and w_{rp} is a factor we calculate based on the collected information:

$$w_{rp} = 1 + \alpha f\left(Rank_{rp}\right) - \beta T_{rp} - \gamma P_{rp} - \delta A_{rp} \tag{3}$$

where $f\left(Rank_{rp}\right)$ depends on the position, if any, of reviewer r in the list of proposed reviewers by the authors and T_{rp}, P_{rp}, A_{rp} are indicator variables described in Table 1.

We select constants α, β, γ and δ by trying different combinations and subjectively evaluating the results based on our experience, until we find a set that produces overall good assignments. (Ideally, however, these parameters should be tuned using data collected from previous editions of the conference and community feedback. The same holds for selecting a linear model for combining the factors in Equation 2: we use a simple model, but collecting previous conference data allows for more informed decisions.) Using Equation 1, we produce the meta-reviewer assignments and then go over them to detect and correct corner cases. For the meta-reviewer assignments we only consider reviewers from core program committee.

Symbol	is 1 if r and p do ... otherwise it is 0
T_{rp}	differ in system/non-system trait
P_{rp}	differ in registered primary areas
A_{rp}	have no common registered areas

Table 1: Indicator variables used in modeling

2.4 Reviewer assignments

For reviewer assignments, we use a similar process as for meta-reviewer assignments but now we consider both core and regular program committee members. For starters, we mark meta-reviewers also as reviewers of their assigned papers. For regular reviewers, we solve once again Equation 1 but this time with r going over the set of light reviewers and R set to the target number of reviews per paper (minus one, as the meta-reviewer is providing one of the reviews). Also, for regular program committee members, S_{min} and S_{max} change due to the higher number of regular program committee members, compared to core ones.

As in the first submission cycle we solicited reviewer suggestions from the meta-reviewers, an extra factor in the calculation of w_{rp} incorporates the suggestions into the formula. Finally, we readjust the weights to tune the relative importance of the features. As both the area and the system traits are input to the process, the set of reviewers per paper may be a mix of both systems and non-systems reviewers. For example, a paper on data cleaning which evaluates the performance of a system will be assigned to both systems and non-systems reviewers who work in the field of data cleaning.

2.5 Timeline

The reviewing process of each submission cycle is divided into four phases as shown in the Gantt chart of Figure 1.

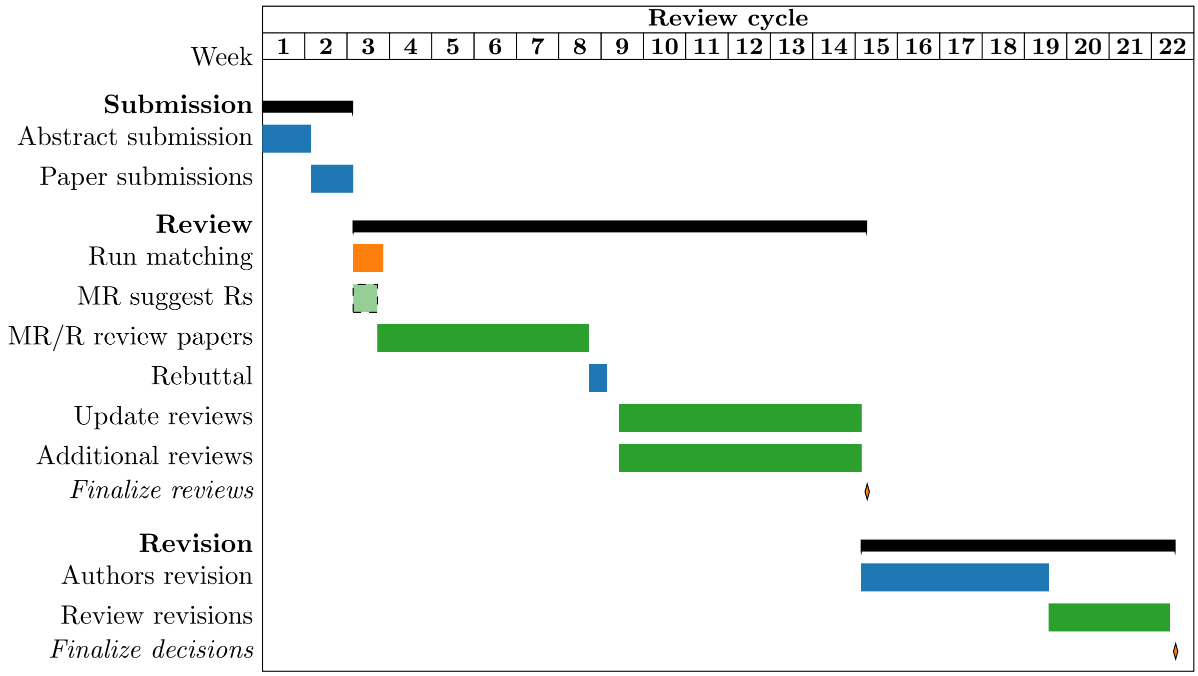

Figure 1: Approximate timeline of reviewing the submissions of each research submission cycle

We augment the submission form with a mandatory field for the authors to characterize their paper as system or non-system, and with an optional field to supply a list of suggested reviewers. Before the submission deadline, we ask the reviewers to upload a sample of their work on TPMS and we manually annotate reviewers as systems or non-systems.

After the submission deadline, we run TPMS to obtain the scores and solve the optimization problem 1, using the collected information to create the meta-reviewer assignments. Then, we manually review the assignments to find sub-optimal assignments and fix them using author suggestions, or secondary matches generated from TPMS. If no reviewers are found, we solicit suggestions from the research community. Afterwards, we upload final reviewer assignments to CMT and launch the review phase. We allow two days for reviewers to express concerns about their eligibility to review papers, and reassign the (typically very few) papers.

Author feedback. Review time is three to four weeks. During that time each reviewer (including meta-reviewers) reads all papers assigned to them and writes their reviews independently. At the end of the reviewing period (plus a few days waiting for late reviews to come in), meta-reviewers initiate discussions. At the same time, we invite author feedback in order to avoid misunderstandings of the text and clarify factual errors. We allow 72 hours for authors to read the (draft) reviews and provide short feedback on factual errors; the time is more than sufficient considering that clarifying any factual errors in the reviews should not take more than a few hours (author feedback is not a revision). In addition, we allow authors to express sensitive issues about the reviews through a new field in the feedback form for specific reviewer complaints which is only visible by the chairs. The meta-reviewers (and chairs) ensure that author feedback is taken into account in the discussions, and the chairs inject themselves into the discussions when a red flag is raised from the authors (through the feedback or via email) or from the reviewers, or when there is a significant disagreement among the reviewers (typically chairs are alerted to such cases by the meta-reviewers).

Assigning additional reviews. If a paper is headed for rejection and at least one reviewer has rated it “weak accept” or more, we invite two additional reviews. We select the additional reviewers as before, i.e., using author suggestions, or secondary matches generated from TPMS, or suggestions from the research community. Reviewers have a one-to-two week time period to review the (typically very few, if any) additional papers.

Review refinement. After all reviews are submitted, reviewers discuss and converge to accept or reject the paper or invite a revision, and the meta-reviewer summarizes the rationale for the decision and revision points (if applicable). An important step before releasing the reviews and meta-review is to carefully polish the reviews to consider author feedback and discussion points, as well as to ensure that all reviewers agree with the meta-review.

Revision. Authors of papers that have been invited for a revision have approximately a month to submit the revised manuscripts along with a letter detailing the main revision points. The reviewers have a couple of weeks to ensure that the revised papers address all points in a satisfactory way. The final decision is to accept or reject the paper. If a paper addresses the revision points but there are still some less significant issues to be fixed, we accept the paper with shepherding (with one of the reviewers acting as a shepherd and communicating with the authors directly). The authors send an updated version of paper to the shepherd a few days before the camera-ready deadline, and the shepherd ensures that all additional points have been addressed and clears the paper for inclusion in the proceedings.

Submission cycles. There are two submission cycles in SIGMOD 2019, one with deadline in July and the other with deadline in November. The described process was repeated for each cycle.

2.6 Implementation

We use CMT as the conference management tool and both the reviewers and authors use this platform. To apply our reviewer assignment process, we extract the required information from CMT and submit our queries for TPMS scores using CMT-TPMS integration.

The integration with our implementation is done by downloading the corresponding files from CMT and incorporating our own files, such as the AMiner generated COI list and our systems/non-systems annotated list of reviewers. We query files with multiple formats in order to generate the $\hat{s}_{rp}$ matrix, as CMT exports almost each part of the data in a different format: xml, tsv, csv, lists per tuple (e.g., list of authors in a submission) and semi-structured text (e.g., author suggestions for reviewers). Then, we solve Equation 1 using CVX [5] and convert the solution to assignments. Finally, we create different views of the assignments and share them with program chairs to inspect a subset of the assignments, excluding from their views and redistributing among them conflicting papers. For simplicity, we flush views into spreadsheets.

Any changes are introduced back to our tool. We use $refinement_{rp}$ to lock assignments across iterations as well as replace some of them. As allowing the integer optimization program to re-run after small manual changes of the weights can produce ripple effects and thus require to recheck every assignment, currently we only use $refinement_{rp}$ for adding and removing assignments rather than changing a weight.

The final assignments are converted into XML files and uploaded to CMT.

3. REWARDING OUTCOMES

Setting up and operating the SIGMOD 2019 reviewing infrastructure for the research track was an end-to-end exciting and rewarding experience. Here we briefly report on the rewarding outcomes and some issues which would make future realizations of our methods a lot easier.

- The number of reviews and average expertise level per paper were improved. We had 3.36 reviews per paper on average (4.64 reviews per rejected paper with at least one weak-accept). Average expertise level[1] per paper was 67.4%.
- Systems and non-systems papers had the same acceptance rate (22.6% vs 20.1%, respectively; calculated over papers of the same trait).
- Reportedly, reviews were of generally higher-quality. Figure 2 shows that 77% of the reviews were made by knowledgeable (50%) or expert (27%), on the topic, reviewers. Only 2% of the reviews had low confidence. Lastly, the distribution of expertise level across system and non-system papers was almost identical, (within a 5% difference).
- Having meta-reviewers actually read and review the papers ensures that the core committee members are aware of the technical details and are involved in discussions more actively than if they are engaged at the end of the review period.
- Establishing author feedback, revisions, and shepherding improves two-way communication between reviewers and authors significantly.
- Allowing authors to express complaints confidentially prevents and defuses subsequent author-reviewer conflicts.

[1]Where 0%, 33%, 66% and 100% correspond to reviewers claiming to be unfamiliar, somewhat familiar, knowledgeable and experts, respectively, with the paper's topic

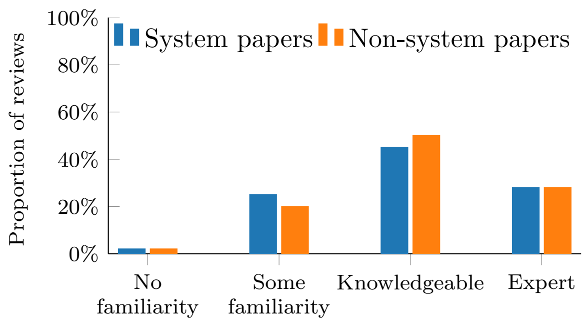

Figure 2: Distribution of review expertise level, as reported by reviewers, over papers in the same author-annotated category (system/non-system). Non-system papers were twice as many as system papers.

4. POINTS FOR IMPROVEMENT

We list the issues which need to be addressed in future editions of the process.

- The most serious time sink was the solicitation of late reviews and trying to reach non-responsive reviewers, through multiple reminders sent via CMT and personal email. Even a handful of late reviewers throws the process off, while non-responsiveness is extremely problematic.
- Some papers are borderline between systems and non-systems; some authors mis-classified their papers. Maybe an option to select "in-between" would have mitigated the problem.
- People change affiliations and thus emails often, which causes problems as CMT uses email addresses as IDs.
- We had to ask CMT many how-to questions. CMT responds swiftly within one to two days. Nevertheless, updated CMT documentation and a more accessible API would facilitate tasks enormously.
- Reviewers need to be educated to provide full citations and to be careful with asking for comparisons with non-peer-reviewed publications such as ArXiv papers. Reviewers should definitely alert authors to such publications, but asking for a head-to-head comparison against recent, non-peer-reviewed work is typically unfair to authors and needs strong justification.
- Re-submissions: If a reviewer receives the same submission that they reviewed for a prior conference, they should be able to mark the paper as a re-submission (visible only to the Chairs to avoid biasing other reviewers). This flag gives the chairs options to appropriately act on it. For example, the chairs might (1) assign an additional (un-)biased reviewer or, (2) in case the paper really has not changed, they may allow the reviewer to resubmit the same review. Another idea is to allow the author to submit the previous reviews with the paper including a description of how they addressed the reviews. This might help papers that are highly controversial and might give the reviewers additional points of view.
- TPMS scores generally worked well, but there were definitely papers and reviewers for which those scores were way off. More specifically, we observed that:
  - The program chairs had to manually check all the reviewer-provided papers in TPMS and manually upload papers for reviewers who did not do it on time. The very helpful CMT and TPMS teams make it easy to send papers from CMT to TPMS and retrieve scores and proposed assignments. Nevertheless, synchronizing the reviewer emails between the two systems and checking whether reviewers have uploaded papers is done through email correspondence with the TPMS team.
  - For some reviewers, the scores result repeatedly in clearly sub-optimal assignments. TPMS works off of a set of representative papers that are uploaded by the reviewers. However, some reviewers upload papers which misrepresent their expertise. We had at least 2 cases for which the assigned papers were far outside the expertise because of that issue. We fixed those cases manually.
  - TPMS logic is counter-intuitive at times: a senior researcher who works on five different topics in depth may upload two papers for each topic. By contrast, a junior researcher who works on one of these topics uploads ten papers on that topic. TPMS appears to give a higher matching score to the junior researcher. The consequences are only mitigated through manual check and adjustment.
  - TPMS can be gamed through rare keywords and it seems trivial to build a tool that tries to significantly increase the

chances of getting certain reviewers from the committee even without their knowledge. While this shouldn't be problematic in practice as it is very similar to explicitly listening "preferred" reviewers, it is something to be aware of.

- The formula we used to combine the "systems" flag and TPMS scores sometimes results in a sub-optimal assignment. For example, consider a systems paper on semantic data integration and two reviewers, one working on the theoretical aspects of semantic data integration and another building systems focused on data cleaning (a topic relevant to, but different than data integration). The first reviewer scores high on topic relevance and the second scores high on the system trait. It is not clear which of these two reviewers is preferable and it becomes even worse considering that the "relevance" and the research area aspects are neither discrete nor well-defined values. This is unfortunately a fundamental issue – it is just not clear how these two indicators should be properly combined to generate a quantifiable "appropriateness" measure. We resolved the issues by checking the assignments ourselves visually as well as asking the reviewers for confirmation on the appropriateness of the assignments and correcting as needed.
- Additional reviews were solicited manually by the chairs and this was a huge time sink, especially when some reviewers refused to take on the additional assignment. The additional review solicitation needs to be automated and reviewer expectations need to be set appropriately beforehand.
- The chairs discovered low-confidence reviews manually; such reviews, however, should be flagged automatically to allow for immediate action.
- A continuous automated analysis of the reviews as they come in to spot problematic text, low-confidence reviews, and poorly attended discussions, could all dramatically alleviate the overhead that chairs and meta-reviewers endure while trying to detect the problem cases manually.
- The meta-reviewers can help suggest appropriate reviewers for papers, but this is only efficient if it is automated.

5. CONCLUSIONS

Overall, the experiment of SIGMOD 2019 with all its changes (TPMS, no reviewer bidding, reviewer suggestions by authors, automatic additional reviewers for papers with one "weak accept", etc.) was a success. Like with every large conference we did receive some complaints, but those were significantly fewer than other years. Furthermore, systems and non-systems papers were treated exactly the same and neither were at a disadvantage. This indicates that SIGMOD should remain a single conference and not be split up as recently suggested [6].

However, there is of course also room for improvements for future conferences and more potential things to try out to improve the overall quality of the reviewing process and with it the quality of the papers.

Automatically adding reviewers for every paper with one "weak accept" is extremely rewarding. In many cases it significantly helped to increase the reviewing quality. Furthermore, reviewing effort is not wasted on papers which are clear rejects/accepts. However, assigning additional reviews has side effects: for instance, the additional reviews may cause the paper to be rejected while without the additional reviewers the paper might have been accepted. This happened with a couple of SIGMOD 2019 submissions and while authors expressed concerns, careful monitoring of the process revealed that the final decision was the correct one: a solid paper can withstand thorough reviewing. As discussed earlier, this process needs to be more streamlined and better integrated into the reviewing workflows.

Preferred Reviewers: Authors have the best understanding of their work and know their peers. As a result they are the experts to resolve misconceptions and propose reviewers for their work. Therefore, increasing author feedback improves the quality of the conference while asking them to suggest reviewers for their work typically provides excellent reviewing matches. Of course, as reviewer identities should be hidden and authors may misuse the feature, great care has to be taken on how the author suggestions are used. Our approach gives low weight to the authors' suggestions during the automatic assignments and increases the weight when chairs are looking for additional reviewers; this reduces the probability of leaking reviewer identities and using malevolent suggestions.

Open reviews: Some conferences have already moved to the open-review concept. While obvious reservations exist (e.g., that a reviewer of a re-submission looks at the reviews of the original rejected submission), open reviewing constitutes a strong incentive to create stronger, more thorough reviews. Thus, testing open reviews for SIGMOD would be an interesting experiment.

Feedback: A rewarding decision is to allow authors to complain about specific reviews. The chairs considered every complaint carefully, investigated them personally, and followed up by discussing with the reviewers and by inviting additional reviews. Several authors expressed their appreciation about the additional steps the SIGMOD organization took to alleviate all misunderstandings.

Recruiting and tracking reviewers: Recruiting reviewers is a manual and slow process, the lack of constructive feedback towards the reviewers and of proper guidelines in writing reviews is still an issue, and external reviewer load and unresponsiveness can have a ripple effect on increasing the load of other PC members. This entire process has to be reconsidered and automated as much as possible.

Reviewer delays Reviewers are typically expected to complete their assignment over several weeks and stage the work. In practice, however, few reviewers are organized enough to distribute their work evenly throughout the review period; most review their entire set of assigned papers in the last few days before the deadline. As is expected, when the deadline arrives several papers are missing reviews and time is wasted trying to hunt down delinquent reviewers and obtain guarantees for delivery of reviews. The reviewing system can provide an optional parameter setting to enable the reviewers to plan their work. The parameter can enable fine-grained deadlines for reviews. For example, if a reviewer is responsible for 20 papers, they can set the deadline for the first five papers in two weeks, the next five after four weeks and so on, and receive automatic reminders. The PC chair may enforce such fine-grained deadlines and receive input on progress. Based on the input, the chair can re-balance load among reviewers as needed.

Plug-in Structure: If existing conference management tools were providing APIs for integrating external plugins/tools and potentially a sandbox environment, many tasks we had to perform would be much easier and faster to do.

Evaluation: The conference system can provide custom success metrics defined based on the procedures in place, and collect data accordingly. For example, it would have been great to know how many reviewers changed their rating after the discussion or after reading author feedback, and how they changed it (favorably or not).

To implement the suggestions above it is important that the PC Chairs are engaged two years before the conference (one year before the first deadline) so that they have time to work with the conference system to put all processes in place.

6. ACKNOWLEDGMENTS

We wholeheartedly thank the authors of SIGMOD 2019 submissions as well as the reviewers for their impeccable collaboration and timely response during the process. We also thank the SIGMOD 2019 executive committee for their support and feedback.

7. AUTHORS AND ROLES

The four authors worked together from December 2017 until June 2019. Anastasia Ailamaki is the SIGMOD 2019 Program Chair, and Amol Deshpande and Tim Kraska are the vice Program Chairs for the SIGMOD 2019 Research Track. The three collaborated with the reviewers in shaping the program of SIGMOD 2019. Periklis Chrysogelos supported the entire reviewing assignment system by designing the formulas, implementing the scripts and tuning and interacting with CMT as needed.

8. REFERENCES

[1] Aminer. `https://aminer.org/`.
[2] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.
[3] L. Charlin and R. Zemel. The toronto paper matching system: an automated paper-reviewer assignment system. *Proceedings of the 30th International Conference on Machine Learning*, 2013.
[4] Conference management toolkit. `https://cmt3.research.microsoft.com`.
[5] M. Grant and S. Boyd. Cvx: Matlab software for disciplined convex programming, 2014.
[6] M. Stonebraker. My top ten fears about the DBMS field. In *34th IEEE International Conference on Data Engineering*, 2018.
[7] C. J. Taylor. On the optimal assignment of conference papers to reviewers. *Technical Report*, 2008.