Reviewing the Systematic Review

William Wallis — 2025138W

December 19, 2016

Abstract

Systematic reviewing is a technique for bringing scientific rigour to a computer science literature review, pioneered by Barbara Kitchenham (Kitchenham, 2004). Specifically, Kitchenham's systematic reviews utilise concepts from the field of medical research to create literature reviews which are repeatable, and produce statistical and empirical results. 12 years after Kitchenham's original guidelines were set for structuring a systematic literature review, the technique has seen widespread adoption — but the original guidelines raise questions and note possible issues with the method. With a wide set of samples to choose from, a review of these systematic reviews may highlight whether these concerns are worth revisiting, before Kitchenham's guidelines — or other methods derived from them — become standard practice for the software engineering research community.

1 Introduction

A Systematic Review is a literature review which collates the results of many papers, using statistical analysis to draw empirical results about the state of a research field and to answer research questions posited as the motivation for the review. Systematic reviews are born from the philosophy that a literature review should have scientific merit, and produce reproducible results, rather than standard techniques for literature reviews, which leave more room for subjective insight. The scientific nature of a systematic literature review, in theory, removes ambiguity and bias from literature review practice and lends the review the same validity and credence as a research study.

Kitchenham's systematic literature review technique has begun to dominate as a literature review technique for software engineering. Kitchenham's review procedure stems largely from literature review techniques in medical research (Kitchenham, 2004; Khan et al., 2001), where empirical studies which verify the validity of literature already published in peer-reviewed journals is paramount to civilian safety. While the technique has seen widespread adoption, some issues exist with the implementation — as noted by Kitchenham herself in a systematic review of systematic review procedures (Kitchenham

and Brereton, 2013). More fundamentally, in Kitchenham's original guidelines there exist some notes which cast doubt on the suitability of a systematic review in the field of software engineering. For example:

> In particular, software engineering research has relatively little empirical research compared with the large quantities of research available on medical issues, and research methods used by software engineers are not as rigorous as those used by medical researchers.

In her guidelines, Kitchenham provides types of empirical data which software engineering research *does* produce which can be appropriate for analysis in a literature review. However, whether research rigour and types of data collected are make appropriate note of by the research community is clear only now that a wealth of systematic literature reviews have been produced.

In this review, a series of systematic literature reviews will be analysed and searched for their scrutiny of research rigour and format of empirical data. In this way, the importance of this doubt regarding the suitability of systematic reviews for software engineering research will be assessed.

The reviews chosen were picked as a result of their popularity on the "*Google Scholar*" academic search engine, found by a search for "software engineering "systematic" literature review", and similar searches. This was to find papers which were well-cited and high-impact, because as the question to be answered would impact the culture around systematic reviews, these papers are important, as they are most likely to influence future systematic reviews.

2 Effect Size Systematic Review

2.1 Systematic Method

First reviewed is a study into effect size in software engineering experiments: Kampenes et al. (2007). The review was conducted according to systematic review guidelines — the guidelines were not specified, however the procedure roughly aligned with that of Kitchenham, and Kitchenham's guidelines were cited as reasoning when defining a search criteria.

As required when developing a systematic review, Kampenes et al. (2007) used search criteria to determine the literature which was to be reviewed. This search criteria was cited from another paper (Sjøberg et al. (2005), which shares some authors), rather than stated directly.

The method for data extraction was also well reported, as a systematic review should entail by Kitchenham's guidelines. The paper goes on to perform a deep and thorough statistical analysis, and concludes with a review of the results of these analyses with a

comparison to the results of similar papers in Psychology and Behavioural Science. The paper succeeded in selecting several papers with empirical data so as to perform the statistical analysis with a large sample.

2.2 Data Extracted

This paper confirmed Kitchenham's doubting note in this case. While a large sample was indeed selected after a time — 92 papers over the course of 10 years — selecting those papers required reviewing the contents of 5453 software engineering papers for suitable results. In the comparison with other fields, the similar Behavioural Science paper found 475 papers with suitable data to review. A similar education literature review found 226, published within a span of a single year.

Of those 5453 experiments — 1.4% of the original sample — only 78 articles actually contained the controlled experiments sought by the authors. When compared to the similar work in other fields, computing science papers were thrice as likely to report effect size as education papers. However, education papers with suitable empirical experiments were published almost thirty times more frequently. Controlled experiments are therefore significantly less readily available than in other subjects where systematic reviews are a suitable method of literature review.

This does not dictate that systematic reviews should not be carried out in software engineering — it does suggest, though, that empirical data might not be very generally available. It would therefore require researchers to wait a large span of time for suitable quantities of data to be produced to create a systematic review.

3 Global Software Engineering

3.1 Intent of Work

Šmite, Wohlin, Gorschek, Feldt, Šmite, Wohlin, Gorschek, and Feldt (2010) sets about the task of reviewing literature on global software engineering (GSE). Particularly, it attempts to collate and assess the results of literature which produce empirical data. The authors identify that there exists scarce literature on the topic, and so to collate the findings and categorise the growing yet important field, they employ a systematic review as a technique for categorising literature based on emerging trends.

The review guidelines used were from a Kitchenham-like standard (Kitchenham and Charters, 2007). The authors do not give a justification for the use of a systematic technique as opposed to a regular review. However, they do note that no systematic review yet existed — so one is inclined to suppose that the authors sought to fill the niche they had identified. Šmite et al. do present a useful section explaining their search method, which they admit as broad.

3.2 Suitability of Systematic Approach

In the paper's abstract, the authors claim that:

> ...the systematic review results in several descriptive classifications of the papers on empirical studies in GSE and also reports on some best practices identified from literature.

(Šmite et al., 2010)

This is true. However, the review also fails to produce useful data as due to "...the limited amount of data, statistical analysis was infeasible". Due to the lack of statistical analysis, the results of the paper may as well have been produced by an ordinary review with specific search criteria. The systematic review process itself was useful in highlighting the need for research questions and search criteria, but the results born of the research did not produce the quantitative data a systematic review's value derives from. The same results could have come from a paper with no systematic requirement, but which borrowed a few techniques from systematic practice.

It is worth noting that, in this review, roughly 18% of the literature found was suitable for the research at hand. These statistics fare much better than those of Kampenes et al.. However, the original set of papers was less than 10% of Kampenes et al.'s set. Again in this case, systematic reviews eschew semantic insight for the repeatability of empirical study, but insufficient data exists to reliably and repeatedly carry out these review experiments.

4 Paper 3

5 Paper 4

6 Paper 5

7 Paper 6

8 Conclusion

References

Vigdis By Kampenes, Tore Dybå, Jo E. Hannay, and Dag I K Sjøberg. A systematic review of effect size in software engineering experiments. *Information and Software Technology*, 2007.

Khalid S Khan, Gerben Ter Riet, Julie Glanville, Amanda J Sowden, Jos Kleijnen, et al. *Undertaking systematic reviews of research on effectiveness: CRD's guidance for carrying out or commissioning reviews*. Number 4 (2n). NHS Centre for Reviews and Dissemination, 2001.

Barbara Kitchenham. Procedures for Performing Systematic Reviews. 2004.

Barbara Kitchenham and Pearl Brereton. A systematic review of systematic review process research in software engineering. *Information and Software Technology*, 55(12): 2049–2075, 2013.

Barbara Kitchenham and Stuart Charters. Guidelines for performing Systematic Literature Reviews in Software Engineering. Technical report, 2007.

Dag I K Sjøberg, Jo E. Hannay, Ove Hansen, Vigdis By Kampenes, Amela Karahasanović, Nils Kristian Liborg, and Anette C. Rekdal. A survey of controlled experiments in software engineering. *IEEE Transactions on Software Engineering*, 31(9):733–753, 2005.

Darja Šmite, Claes Wohlin, Tony Gorschek, Robert Feldt, D Šmite, C Wohlin, T Gorschek, and : R Feldt. Empirical evidence in global software engineering: a systematic review. *Empir Software Eng*, 15:91–118, 2010.