Information and Software Technology xxx (2009) xxx–xxx

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

Empirical evaluation in Computer Science research published by ACM

Jacques Wainer [a,*], Claudia G. Novoa Barsottini [b], Danilo Lacerda [a], Leandro Rodrigues Magalhães de Marco [a]

[a] Computing Institute, University of Campinas, Av. Albert Einstein 1251, 13083-852 Campinas, SP, Brazil
[b] Department of Health Informatics, Federal University of Sao Paulo, Brazil

ARTICLE INFO

Article history:
Received 2 September 2008
Received in revised form 5 January 2009
Accepted 18 January 2009
Available online xxxx

Keywords:
Empirical evaluation
Research evaluation
Experimentation
Systematic review

ABSTRACT

This paper repeats part of the analysis performed in the 1995 paper "Experimental evaluation in Computer Science: a quantitative study" by Tichy and collaborators, for 147 papers randomly selected from the ACM, published in the year 2005. The papers published in 2005 are classified in the following way: 4% theory, 17% empirical, 4.7% hypothesis testing, 3.4% other, and 70% design and modeling (using the 1995 paper categories). Within the design and modeling class, 33% of the papers have no evaluation. The numbers of the 2005 sample are very similar to the original figures for the 1995 sample, which shows that Computer Science research has not increased significantly its empirical or experimental component.

1. Empirical evaluation in Computer Science

Tichy and collaborators [15] evaluated 400 articles published in 1993, 50 of them randomly selected papers published by ACM in 1993 and the rest systematically selected from a few journals in Systems and Software Engineering, and classified the research reported in the paper in five categories (quoting [15] definitions):

- *Formal theory*: articles whose main contributions are formally tractable propositions, e.g., lemmata and theorems and their proofs.
- *Design* and modeling: systems, techniques, or models, whose claimed properties cannot be proven formally. Examples include software tools, performance prediction models, and complex hardware and software systems of all kinds. The papers in this class were further classified in the categories 0%, 0–10%, 10–20%, 20–50%, and +50%, according to the proportion of the paper that was dedicated to the evaluation of the new system, technique, or model.
- *Empirical work*: articles that collect, analyze, and interpret observations about known designs, systems, or models, or about abstract theories or subjects (as this paper does). The emphasis is on evaluation, not on new designs or models.
- *Hypothesis testing*: articles that define hypotheses and describe experiments to test them.
- *Others*: articles that do not fit any of the four categories above, e.g., surveys.

Tichy and collaborators found that for the random sample 12% of the articles were in the theory class, 70% were design and model, 2% were empirical, 2% were hypothesis testing, and 14% were classified as others.

Tichy was particularly worried that from the random sample, 40% of the papers which propose a new system, model, framework, and so on (the design and modeling class), lacked any evaluation (0% of space dedicated to the evaluation). That proportion was 50% for the papers in the Software Engineering journals. In comparison, the papers published in the journals Optical Engineering and Neural Computation, which were used as examples of Engineering journals, only 15% and 12% of the papers that proposed new ideas lack the required evaluation.

According to Tichy:

The low ratio of validated results appears to be a serious weakness in Computer Science research. This weakness should be rectified for the long-term health of the field.

The paper ends with a call for better standards in performing and publishing Computer Science (CS) research, with a greater emphasis on the evaluation of claims and designs, and a less emphasis on the creation of new systems, models, algorithms, frameworks, and so on.

This paper tries to evaluate if the Computer Science community has answered Tichy's calls. In particular, we repeated the evaluation of 147 randomly selected from all papers published by the ACM, including ACM journals, conferences, and workshop, and available in the ACM digital library, for the year 2005.

We understand that there are two main criticism to using the ACM published papers as representative of the whole CS research. The first one is that not all Computer Science sub-areas are well

* Corresponding author.
E-mail address: wainer@ic.unicamp.br (J. Wainer).

0950-5849/$ - see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.infsof.2009.01.002

represented in ACM conferences and journals. Artificial Intelligence is one of such areas. The second main criticism is that most of the research published in by ACM are conference and workshop papers, which are usually short, and describe not fully developed ideas. One would expect journal papers would contain a more substantial evaluation of the first ideas presented at a conference or workshop.

We agree with both criticisms, and we agree that our sampling has bias, but we feel that there is no better alternative as a sample of CS research. The main problem is *selection bias* – any inclusion of a *particular* journal or set of journals, or non-ACM conferences into the analysis can be criticized as distorting the conclusions toward a particular direction. For example, the inclusion of papers from the non-ACM journal "Empirical Software Engineering," whose papers have probably a high content of empirical research, would only be meaningful if that journal is in some way "representative" of the whole of CS journals. We do not know *a priori* of any journal that is "representative" of CS research.

The only solution to avoid the selection bias is to evaluate a *random sample* from some "representative" *sub-population* of the whole set CS research. In the case of this paper, the sub-population was all the 2005 ACM published papers. As we discussed above, this is not a totally "representative" sub-population, but we do not believe that there is a better one. Some alternatives for the sub-population are: IEEE published papers (for the year 2005), or the set of papers published in journals indexed by the Web of Science, or journals indexed by Scopus, or papers indexed by the DBLP.

Of course, we have no data to support our claim that among these possible sub-populations, the ACM published papers is the "most representative." The Web of Science or Scopus indexed sub-populations are solely or mainly of journal and it seems clear that most of the CS production is published in conferences and workshops. The IEEE sub-population seems to us even more bias towards some of CS sub-areas than the ACM. Finally we do not know how to evaluate the bias of the DBLP indexing service.

If the reader does not agree with our intuitions that the ACM is the "most representative" sub-population of CS research, then this paper should not be read as an evaluation of the CS field as a whole, but only as an evaluation of the ACM published papers.

Finally, the decision to use Tichy's taxonomy for classifying the empirical content of a paper was forced on us by the goal of replicating the 1995 evaluation. There are some terminological problems with the taxonomy. The major one is the name of the "empirical work" class. Tichy uses "experimental" research as the term that describes research that gathers data from "the reality". We agree with [13] that "empirical" research is a better term. Experiments are a particular form of empirical research. Thus Tichy's class of "empirical work" is not the whole or the central component of empirical research, it is just one form of empirical research in which the author is not proposing any new system, model, framework, and so on but evaluating an existing one. Another problem is that Tichy's class of "other" does not distinguish between some forms of empirical research such as surveys and secondary or bibliographic research such as this one, and other types of publications that are not empirical at all.

1.1. Related results

Glass and collaborators [6,12] analyzed 628 Computer Science papers from 1995 to 1999 and classified them regarding different dimensions. Of interest to this paper is the "research approach" dimension, whose possible values are descriptive research, formulative research or evaluative research. The work [6] found that 9.88% of the research was descriptive and 10.98% was evaluative and 79.15% was formulative.

If one considers that design and modeling class used in this paper, corresponds at least in part, with Glass's formulative approach to research, then one can conclude that from 70% to 80% of the published results in CS from 1993 to 1999 are mainly the proposal of a *new* system, model, algorithm, process, taxonomy, framework, and so on.

In specific CS domains, Prechelt [11] evaluated 190 articles on neural network published in 1993 and 1994. The author discovered that only 8% of the papers presented results on more than one realistic data set, and 1/3 of the papers had no quantitative comparison to previously known algorithms.

In Computer Supported Collaborative Work (CSCW), Pinelle and Gutwin [10] classified all papers published in the ACM CSCW conferences from 1994 to 1998 regarding the goals and methods of evaluation of the systems proposed. Wainer and Barsottini [16] classify all the papers published in the ACM CSCW conferences from 1998 to 2004 into categories similar to the ones used by Tichy et al. [15], whether the paper proposes a new system (and how it is evaluated), whether it tests an hypothesis, whether it describes an existing collaborative work/entertainment environment, or whether the paper has no empirical component (or in terms of [15], it is a design and project with 0% of evaluation).

Software engineering, and in particular *empirical* software engineering is an area in which bibliographic (or secondary) research seems more frequent. Kitchenham and collaborators [9] analyzed the use of systematic literature review in empirical SE since 2004. They found 20 of such studies after reviewing 10 journals and 4 conferences in the area. Glass and collaborators [7] used the same dimensions of analysis as in [6,12] to classify 369 journal papers in SE published in the period 1995–1999 randomly selected. Again, using the "research approach" dimension, they concluded that 55% of the papers followed the "formulative" approach. Zelkowitz [19] followed up on a previous research [18] and compared the evaluation methods used in papers published in one SE conference and two SE journals, from 1985 to 2005 (in 5 years intervals). Zelkowitz used a 14 classes categorization (as opposed to the five classes used in this paper), and found that papers without any empirical validation when one was possible (which corresponds to our design and modeling with 0%), dropped from 27% from 1985 to 16% in 2005. And papers with a weak form of validation (called assertion in [19]) dropped from 35% in 1985 to 19% in 2005. Hofer and Tichy [8] analyzed the papers published in the Empirical Software Engineering journal from 1997 to 2003 and classified them according to topic of research, evaluation methods, and sources of data. Regarding evaluation method, they found that 38% of the papers used an experiment as the empirical research method, 29% used case research, followed by other methods. No temporal analysis was performed. Brereton and collaborators [2] analyzed all the papers from 1993 to 2002 in nine journals and three conferences, and found out that only 1.9% of the articles reported some form of controlled experiment, and found no trend over the years.

2. Method

We followed, to the best of our abilities, exactly the methods used by Tichy et al. [15]. The categories described in that work: formal theory, design and modeling, empirical work, hypothesis testing, and others, are called *major categories*. The divisions within the design and modeling were called *minor categories*.

The method followed in this paper, in a simplified way, was:

(1) CGNB, DL and LRMM read all the 50 articles evaluated in [15] and discussed the classifications attributed in [15] in order to understand how to classify the papers in the major categories.

Please cite this article in press as: J. Wainer et al., Empirical evaluation in Computer Science research published by ACM, Inform. Softw. Technol. (2009), doi:10.1016/j.infsof.2009.01.002

(2) We randomly selected 200 articles published by ACM and available in the ACM portal.[1] The selection of the articles was performed in October of 2006.
(3) From the 200 selected files, we removed the ones that were not in scientific articles (editorials, news, table of content, and so on). We obtained 147 scientific articles.
(4) Each of the 147 articles were attributed to two reviewers among CGNB, DL and LRMM.
(5) Each of CGNB, DL, and LRMM evaluated around 100 articles independently using [15] major and minor classifications.
(6) Articles with the same major classification by its two reviewers were considered closed.
(7) Articles with diverging major or minor classifications, or articles for which at least one of the reviewers was not sure of his/her own classification, were discussed by its two evaluators. If there was no agreement between them, JW would also evaluate it and either the most common class was attributed to the paper, or if there were three different classifications, JW's classification would be attributed to the paper.

The main difference from our methodology to Tichy's is that we decided on using only two reviewers for each paper, whereas in Tichy's paper, all four authors reviewed the random sample set, but the final results reported refer to the classification of a single reviewer. The other reviewers classification was used to measure the classification error (of the reported results due to a single reviewer). In our case, we used at least two evaluations to reduce the subjectivity of the classification. Second, we started with independent classifications, but if there was a disagreement between the reviewers, we allowed for a joint discussion in an attempt to reach a final classification. Again, the goal was to reach a classification with some control on the subjectivity.

CGNB, DL, and LRMM are graduate students with different experiences with the CS literature. DL and LRMM are more experienced with image processing literature, and CGNB, with medical informatics. JW is a tenured faculty with 15 years of experience in CS areas such as Artificial Intelligence, Medical Informatics, and Computer Supported Collaborative Work.

2.1. Random selection

We decided on selecting a sample from the papers published in 2005 because, by October of 2006, when this research started, it was unclear how many of the papers published in 2006 were already available in the ACM digital library.

The articles were selected using the following procedure. From the ACM Portal page, we navigated to the advance search for ACM Digital Library (not the ACM Guide to the Computer Literature), and used the following search criteria: *Published since January 2005, Published before December 2005*, and *any type of publication*. The search, made in October 2006, returned 16,162 articles.[2] We wrote a program that collects the pdf links in all the pages of the search result. A second program randomly selected 200 entries from the list of links, and downloaded the corresponding pdf.

2.2. Number of articles selected

The number of articles selected was determined by the number of articles [15] used. Using the same query specified in Section 2.1 for 1993, we discovered that 5095 articles from 1993 are available from the ACM digital library. If we consider that this is the total number of articles published by ACM in 1993, then Tichy reviewed around 1% of the ACM papers of that year. We followed that same heuristics, and given that the query to the ACM digital library described above, returned 16,162 articles published in 2005, we decided to randomly select 200 articles.

After a first analysis, we removed 53 of them because they were not scientific articles, but publications like table of contents, abstracts, editorials, news, and so on. Although the final number of selected papers (147) is smaller than our goal of 1% of the total number of published papers in the period, we decided not to add any more papers.

2.3. Confidence intervals

The confidence intervals are all at a confidence level of 90%. We use the adjusted Wald confidence interval for proportions [1]. Given that from a population of n, x of them are "success", the adjusted Wald interval computes the proportion of successes as

$$p = \frac{x + z_\alpha^2/2}{n + z_\alpha^2}$$

(instead of the standard proportion x/n) and compute the confidence interval using the standard Wald formula

$$\mathrm{CI} = p \pm z_\alpha \sqrt{\frac{p(1-p)}{n + z_\alpha^2}}$$

where z_α is the z-critical value for the α confidence level.

3. Results

The technical report [17] lists each of the 147 papers and their classifications. Of the selected papers, 13 were journal papers, two were published on SIG bulletin, and the rest was conference papers. Table 1 lists the results of the classification of the 147 papers.

3.1. Difficulties

The main problem with this form of bibliographic research is the subjectivity of the classification. For 90 of the 147 papers, the two original reviewers, independently, arrived to the same classification.

Of the remainder 57 papers for which there was no independent agreement, 42 were resolved by the discussion of the two original evaluators. Among these 42, 15 (36%) were divergence between adjacent minor classifications, for example, one would classify the paper as design and modeling with 10–20% evaluation, and the other design and modeling, with 20–50% evaluation. Another 9 (21%) were both classified as design and modeling, but the minor classifications were not adjacent. Finally, of the 42, 18 (42%) were classified independently in different major categories.

Table 1
Totals for the classification of the 147 papers.

Class	Number	Percentage of total	Percentage of design
Theory	6	4	
Empirical	26	17	
Hypothesis	7	4.7	
Other	5	3.4	
Design total	103	70	100
0%	34	23	33
0–10%	10	6.8	9.7
10–20%	22	14	21
20–50%	31	21	30
>50%	6	4	5.8

[1] portal.acm.org.
[2] At the time of the writing of this article, the same search returns 16,188 articles.

Please cite this article in press as: J. Wainer et al., Empirical evaluation in Computer Science research published by ACM, Inform. Softw. Technol. (2009), doi:10.1016/j.infsof.2009.01.002

Table 2
Confusion matrix for the 57 papers with some divergence of classification.

	0–10%	10–20%	20–50%	>50%	Empirical	Hypothesis	Other	Theory
Design 0%	1	12	1	0	7	5	5	0
Design 0–10%		6	3	0	1	0	0	1
Design 10–20%			11	0	2	1	2	0
Design 20–50%				8	2	0	0	0
Design >50%					1	0	0	0
Empirical						1	4	0
Hypothesis							2	0
Other								0

Table 3
Ninety percentage confidence interval (using the adjusted Wald method) for the proportions (in percentages) for each class, for 1993 and 2005.

Class	1993	2005
Theory	6.1–21.8	2.0–7.8
Empirical	0–9.1	13.1–23.5
Hypothesis	0–9.1	2.4–8.7
Other	7.6–24.1	1.5–6.9
Design total	58.5–79.4	63.5–75.8
0%	20.5–41.5	17.9–29.3
0–10%	2.0–15.5	4.0–11.1
10–20%	6.1–21.8	10.7–20.5
20–50%	13.8–33.0	16.1–27.1
>50%	0–6.1	2.0–7.8

Fifteen papers required the evaluation from JW. Of these, 4 were divergences on minor adjacent categories, 3 were divergence on minor non-adjacent categories, and 8 were divergence on major categories.

Table 2 is the confusion matrix for the 57 papers for which there was no independent agreement by the two original evaluators. The entry 12 in the line design 0% and column D. 10–20% indicates there for 12 papers there was at least *one* classification as design and modeling with 0% of evaluation and *one* as design and modeling with 10–20%. Of the largest figures in Table 3, the confusion between design 0% and design from 10% to 20% seems due to divergences on what evaluation is – both reviewers agreed that the paper proposes a new entity, but they disagree on whether there is or not evaluation of the entity in the paper. The second largest confusion is between design 10–20% and design 20–50% which seems to indicate differences in each reviewer's methods of calculating the area dedicated to evaluation. Finally, there are a large number of confusions are between design 0% and the empirical and hypothesis classes. That confusion is very puzzling because empirical and hypothesis papers are mainly about evaluation, and a design 0% has no evaluation at all.

4. Conclusions

This research classified 147 randomly selected papers published by ACM in 2005 in 5 classes proposed by Tichy and collaborators [15]. The classes describe different forms of research in Computer Science and in some way different epistemologies of CS.

The theory class reflects a mathematical view of CS, in which the research contribution is mainly in the form of proofs of theorems. The hypothesis testing class reflects a Natural Sciences and Popperian view of CS, in which an hypothesis is clearly stated and an experiment is performed that confirms or disproves the hypothesis. The empirical and the design and modeling classes reflect an engineering view of CS. The design and modeling class includes papers that propose a *new* system, algorithm, program, model, methodology, framework (which we will collectively refer as an *entity*), and optionally evaluates this new entity. The empirical class includes papers that evaluate entities proposed by others. These three epistemologies for CS – as a form of mathematics, as a form of Natural Sciences and as a form of engineering, has been discussed by other authors [3–5].

This paper evaluated how CS is done, or what kind of science Computer Science is, by analyzing what is published as CS research (by ACM in the year 2005).

From the ACM sample, we conclude that Computer Science follows mainly an engineering epistemology – most CS research is the proposal of *new* entities. Our research points that the empirical and the design and modeling classes amount to 87% (17 + 70%) of the papers published by ACM in 2005. Theory papers are 4% of the total and hypothesis testing papers are 4.7% of the total.

But even if one assumes that CS mostly follows an engineering epistemology, the amount of evaluation reported in the papers is low. Of the 129 papers in the "engineering epistemology" class (empirical + design and modeling), 34 (26%) have no evaluation. If we focus on the design and modeling category, a third of the papers that propose a new entity, do not evaluate them at all! If one arbitrarily determines that at least one fifth of the space of paper that proposes a new entity should be dedicated to evaluating it, then only 36% of the papers satisfy this mild requirement.

Tichy et al. [15] evaluated all the papers published in 1993 in Optical Engineering, which he classified as a Engineering journal, and in that sample only 15% of the design and modeling papers had no evaluation at all, and 67% of the design and modeling papers had at least one fifth of the space dedicated to evaluation. Thus the figures for CS published in 2005, regarding evaluation, are worse than they were for an Engineering journal in 1993.

One can argue that the low amount of evaluation is typical of conference papers in CS, and that journal papers would probably have more rigorous standards. In this research we did not try to verify this hypothesis directly, but within our random sample of 147 papers, there is a sub-sample of 13 journal papers, distributed in the following categories: 1 paper in other, 2 papers in empirical, 1 in design 0%, 2 in design 0–10%, 4 in design 10–20%, 2 in design 20–50%, and 1 in design >50%. Thus, in the sub-sample 10% of the design and modeling papers have no evaluation, and 30% have at least a fifth for evaluation. Unfortunately, since the size of the sub-sample is small, the confidence interval for these proportions are large. The difference between the 10% without evaluation published in journal and the global 34% is not statistically significant with 90% confidence.

Thus, if one assumes that the journal Optical Engineering is representative of the Engineering publications (in 1993), then although CS follows in most part an Engineering epistemology, it still falls behind in the rigor of empirical evidence it demands from its research, in comparison with other Engineering domains.

The comparison of the results from 1993 [15] and the results from 2005 shows that the distribution of papers among the different categories has changed very little. Table 3 lists the proportions of each classification for both years, with a 90% confidence interval. Thus, one can only state, with 90% confidence, that the number of empirical papers increased in the period. If one pays less attention to the confidence intervals, one may find a small trend towards more evaluation in design and modeling papers.

The increase of empirical papers is interesting. It shows that there was an increase of *reuse* of other people's research in Computer Science. Empirical papers have an emphasis in evaluating an *existing* program, algorithm, model, framework, and so on. Thus, from 1993 to 2005 there was a small increase in research that evaluates already existing entities.

Please cite this article in press as: J. Wainer et al., Empirical evaluation in Computer Science research published by ACM, Inform. Softw. Technol. (2009), doi:10.1016/j.infsof.2009.01.002

If one agrees with Tichy's call for more evaluation in Computer Science research, in the conclusions of [15] and specially in [14] (as we do), then the comparison of the 1993 results with the 2005s is somewhat discouraging – little has changed since 1993.

To finalize, we would like to speculate on why there was so little change from the 1993 evaluation. If one looks into specific subareas within Computer Science, there seems to be some evidence that the more traditional conferences and journals tend to assume a more empirical minded framework, or in other words, the papers published in these conference and journals tend to have more empirical content. That is the conclusion from Zelkowitz [19] for SE and Wainer and Barsottini [16] for CSCW. We feel that the move towards more empirical evaluation of the research published in the more traditional venues is true for other areas in which the authors are involved such as computer vision, machine learning, and medical informatics. Probably that is true for other areas in CS as well.

We believe that this move towards more empirical evaluation is not vivid in the comparison between 1993 and 2005 because new CS areas and new conferences and journals are added to the set of CS production venues. These new venues and areas are in a development stage in which empirical content is not as strongly required as in the more traditional areas/venues.

This theory suggests some future research. The first one is the verification if other traditional CS areas/venues have this move toward an increase in empirical content of the published papers. A second line of research is the comparison regarding empirical content among traditional and newer venues within the same CS area.

We would like to finalize by pointing out the centrality of the design and modeling (or formulative research in Glass' terms [6,12]) in the practice of CS research community. The community see itself as *creating* new entities. It is possible that in a community that emphasizes creativity so strongly, empirical evaluation and rigor would always be less important values.

References

[1] A. Agresti, B.A. Coull, Approximate is better than exact for interval estimation of binomial proportions, The American Statistician 52 (1998) 119–126.

[2] P. Brereton, B.A. Kitchenham, D. Budgen, M. Turner, M. Khalil, Lessons from applying the systematic literature review process within the software engineering domain, Journal of Systems and Software 80 (4) (2007) 571–583.

[3] P.J. Denning, Is computer science science? Communications of the ACM 48 (4) (2005) 27–31.

[4] P.J. Denning, Computing is a natural science, Communications of the ACM 50 (7) (2007) 13–18.

[5] A.H. Eden, Three paradigms of computer science, Minds and Machines 17 (2) (2007) 135–167.

[6] R.L. Glass, V. Ramesh, I. Vessey, An analysis of research in computing disciplines, Communications of the ACM 47 (6) (2004) 89–94.

[7] R.L. Glass, I. Vessey, V. Ramesh, Research in software engineering: an analysis of the literature, Information and Software Technology 44 (8) (2002) 491–506.

[8] A. Hofer, W.F. Tichy, Status of empirical research in software engineering, in: Basili et al. (Eds.), Empirical Software Engineering Issues, LNCS, vol. 4336, Springer Verlag, 2007, pp. 10–19.

[9] B. Kitchenham, O.P. Brereton, D. Budgen, M. Turner, J. Bailey, S. Linkman, Systematic literature reviews in software engineering – a systematic literature review, Information and Software Technology 51 (1) (2009) 7–15.

[10] D. Pinelle, C. Gutwin, A review of groupware evaluations, in: Proceedings of the WET ICE 2000, IEEE Computer Society, 2002, pp. 86–91.

[11] L. Prechelt, A quantitative study of experimental evaluations of neural network learning algorithms: current research practice, Neural Networks 9 (3) (1996) 457–462.

[12] V. Ramesh, R. Glass, I. Vessey, Research in computer science: an empirical study, Journal of Systems and Software 70 (1–2) (2004) 165–176.

[13] D. Sjoberg, T. Dyba, M. Jorgensen, The future of empirical methods in software engineering research, in: Future of Software Engineering, FOSE 07, IEEE, 2007, pp. 358–378, doi:10.1109/FOSE.2007.30.

[14] W.F. Tichy, Should computer scientists experiment more? Computer 31 (5) (1998) 32–40.

[15] W.F. Tichy, P. Lukowicz, L. Prechelt, E.A. Heinz, Experimental evaluation in Computer Science: a quantitative study, Journal of Systems and Software 28 (1) (1995) 9–18.

[16] J. Wainer, C. Barsottini, Empirical research in CSCW – a review of the ACM/CSCW conferences from 1998 to 2004, Journal of the Brazilian Computer Society 13 (2007) 27–36.

[17] J. Wainer, C. Barsottini, D. Lacerda, L. de Marcom, Experimental evaluation in Computer Science II: a quantitative study, 12 years later, Technical Report IC-09-02, Institute of Computing, University of Campinas, 2009. Available from: <http://www.ic.unicamp.br/~reltech/>.

[18] M. Zelkowitz, D. Wallace, Experimental models for validating computer technology, IEEE Computer 31 (5) (1998) 23–31.

[19] M.V. Zelkowitz, An update to experimental models for validating computer technology, Journal of Systems and Software, in press, doi:10.1016/j.jss.2008.06.040.

Please cite this article in press as: J. Wainer et al., Empirical evaluation in Computer Science research published by ACM, Inform. Softw. Technol. (2009), doi:10.1016/j.infsof.2009.01.002