Guidelines for Application Papers Submitted to PETRI NETS

Kees van Hee[1], Lars M. Kristensen[2], and Laure Petrucci[3]

[1] Department of Mathematics and Computer Science
Eindhoven University of Technology
5600 MB, Eindhoven, NETHERLANDS
k.m.v.hee@TUE.nl
[2] Department of Computer Engineering
Bergen University College
5020 Bergen, NORWAY
lmkr@hib.no
[3] LIPN, CNRS UMR 7030, Université Paris XIII
99, avenue Jean-Baptiste Clément
F-93430 Villetaneuse, FRANCE
laure.petrucci@lipn.univ-paris13.fr

Abstract. This paper provides guidelines for the publication of papers on applications of Petri nets and concurrency. We summarise the lessons learned from validating the guidelines on application papers submitted to the conference in the past. Finally, we provide references to a set of selected application papers intended to serve as inspiration for authors when preparing application papers.

1 Introduction

Although it is generally accepted that "nothing is more practical than a good theory", society is more and more expecting practical results from scientists. Therefore, we need evidence that a theory can be applied in practice and that it contributes to the solution of real-world problems. Application papers that describe case studies are a way to publish this evidence and the PETRI NETS conference (full name: International Conference on Application and Theory of Petri Nets and Concurrency) stimulates the publication of such application papers. The publication of application papers is also important for the industrial adoption of the research results and tools produced by the scientific community.

In the last years, there have been several problems with submitted application papers, and there has been a recurrent discussion in the programme committee on what constitutes a good application paper. Furthermore, many papers submitted as application papers were misclassified by the authors at the time of submission. Many application papers turned out to be in fact rather weak theory papers with a relative large example. Other application papers are describing a real application with a more or less standard solution, which does not provide much new insights or impact. All this has led to a decline in the number of

submitted and accepted application papers over the past years, and part of the problem may be that there are no clear guidelines for a good application paper.

This paper provides guidelines for writing as well as for evaluating application papers for the PETRI NETS conference and is aimed at improving the situation outlined above. The purpose of the guidelines is to help authors and reviewers and should not be considered a straitjacket for writing application papers. Our guidelines concentrate on what a typical application paper should look like, but we do not exclude the submission and subsequent acceptance of application-oriented papers that do not conform to the guidelines. Examples of such papers may include papers describing lessons learned from teaching and technology transfer activities, and papers dealing with standardisation work.

This paper is structured as follows. In section 2, we present guidelines for a paper to qualify as an "application paper" (i.e., the nature of application papers), and provides a set of criteria that can be used when evaluating application papers. Then, the different aspects to be addressed in application papers are stated in section 3 in the form of *content elements*. These guidelines are validated in section 4 on a set of application papers that were accepted for the PETRI NETS conference in the past and also on a set of rejected papers. Finally, in section 5 we provide references to selected application papers that have been published in the past. Throughout this paper we use Petri nets as the underlying formalism being applied, but the guidelines provided also apply to the application of other models of concurrency.

2 Qualification Criteria and Evaluation Guidelines

This section states some basic requirements for application papers. The *first* question to be answered is: "What is an application?". In this context, an application uses Petri nets to obtain results or solutions concerning a real-world problem, or a class of real-world problems.

This question introduces a *second* one: "What is a real-world problem?". A real-world problem (in this context) is concerned with a system or process in the real-world, such as man-made systems for telecommunications, logistics, manufacturing, computer systems or other business processes, and with systems or processes that occur in nature such as studied in physics, chemistry, and biology. Although it is possible to apply Petri nets to some other formalism to prove properties of that formalism, we do not consider that as an application in this context.

A *third* question is: "What kind of problems are we considering?". Since it should be problems that can be solved using Petri nets, the solution consists of two parts; firstly a Petri net model of some real-world system or process is made, and secondly the model is used to answer questions concerning the system.

The following list shows some possible usages of a model which in turn constitute guidelines for a paper to qualify as an application paper:

- *Making a specification* of the system. The model is to be used as a formal specification of system requirements or system design. The system may be an existing man made system, a system in nature or a system to be constructed.
- *Understanding the behaviour* of the system. For instance by animation or simulation.
- *Proving static or behavioural properties* of the system. For instance the analysis of conformance to a set of rules by means of structural analysis or state space exploration.
- *Making predictions* of the behaviour of the system. For instance by mapping the state of the real system to the marking of the Petri net and applying simulation or Markov analysis.
- *Computing performance indicators* of the system. For instance using simulation or Markov analysis.
- *Constructing a real system* using the model as the *blue print* for building the system. For example the model will be translated into executable program code or the model is used as an early prototype of the real system.
- *Controlling the system via the model*. For example the model could be used as a configuration parameter for a workflow engine, or to control a hardware system by enabling/preventing the occurrence of actions.

An obvious requirement for an application to be a *scientific contribution* is that the solution should be *non-trivial*. Even though this is a subjective criterion, it means that the solution should not be the result of applying a standard textbook technique or a solution that could be provided by an average student after a standard course in Petri net modelling. An application paper often describes a case study in which model design is the key issue. In that sense an application paper is a result of *design science*. This kind of research is becoming more established (see [1]). Here a design is a proof of existence of a solution for a problem and a way to discover aspects of the design that would most likely not have been made explicit without the design process.

So *evaluation criteria* for a scientific contribution include:

- *Novelty*. This criterion concerns the new elements of the study. So the number of new results counts and also the size of the discoveries.
- *Complexity*. This criterion concerns the complexity of the system that is modelled as well as the problem to be solved.
- *Generality*. This criterion concerns the possibilities to apply the solution to other situations or even other application domains.
- *Impact*. This criterion concerns the value that the modelling and analysis efforts have had on the system under consideration.
- *Tools and techniques*. This criterion concerns the (software) tools and techniques used. The novelty of the paper could be that the used tools or techniques really worked in practice.

3 Content Guidelines

Section 2 presented the actual characteristics for the work to be a valuable application paper and for qualifying as an application paper. We now concentrate

on the content of the paper so that it clearly presents the expected key points. An application paper should clearly address the following in its content:

1. The *context* of the problem: A description of the environment in which the problem occurs, including the stakeholders, i.e. the persons or organisations involved in the problem, and their background and concerns. For applications conducted in an industrial context or case studies of a particular system, the modeling and analysis efforts (e.g., person-hours) should be discussed.
2. The *problem* itself: A specification of the problem, a motivation explaining why it is important, and a description of the most critical problems and how they were solved. In case the paper is about a class of problems, the application domain should be described here. In almost all cases the problem specification is based on the context description.
3. The *model* with a motivation for the design choices. The presentation of the model must also address the most important assumptions, abstractions, and simplifications reflected in the model. For larger models it often not possible nor desirable to present the entire model (e.g., due to space limitations). In this case, the paper should provide a description of well-chosen and representative parts of the model, and the description should be such that it gives a good impression of the complexity of the complete model.
4. The *calibration* and *validation*. Calibration in case the model has numerical parameters that have to be estimated. Validation establishes the relationship between the model and the real system, such that we can rely on the model to study the real system.
5. The *application* of the model to answer the questions of interest. This could be the analysis of the model, but also the use of the model as a blue print.
6. The *results* and in particular how successful the model and its use were in solving the problem.
7. The *lessons learnt* and the possibilities to apply them in new situations. A new methodology is an example. Lessons learnt may include an evaluation of applied computer tools and techniques, a description of what was gained by the application of Petri nets, and the benefits and drawbacks compared to other approaches and tools.

In table 1 we show how the seven *content elements* of an application paper specified above contribute to addressing the evaluation criteria presented at the end of section 2. A "+" in an entry specifies that the content element contributes to the evaluation criteria, whereas a "-" in an entry specifies that the content element does not directly contribute to the evaluation criteria.

In case the paper is about a class of models for a specific application domain there is often a *meta-model* involved: i.e. some formal framework based on an existing one, but dedicated to the application domain, for example modelling and analysis of electrical circuits or chemical reactions. In these cases the meta-modelling should not get too much emphasis, because the paper otherwise becomes a theory paper. However the meta-modelling often contributes to the novelty. Also the generality and impact could be high in case of a class of

Table 1. Content elements versus evaluation criteria

	Evaluation criteria				
Content element	**novelty**	**complexity**	**generality**	**impact**	**tools**
context	-	-	-	+	-
problem	-	+	-	+	-
model	+	+	+	-	-
calibration	-	-	-	+	-
application	+	+	+	-	+
results	+	-	-	+	-
lessons learnt	+	-	+	-	+

real-world problems. It is also important that papers dealing with meta-models include application to real-life models and not restrict itself to consider only very small models and examples.

We also propose that authors of application papers (whenever possible) make their models (and other computerised artifacts) created for the application electronically available already at the time of submission. This is analogous to how computer tools must be made available for evaluation of tool papers.

4 Validation of Guidelines

In order to evaluate our guidelines, we studied the proceedings of the PETRI NETS conferences from 2000–2010. We classified in total 34 papers as application papers which is 20% of the regular papers (invited papers and tool papers were not included). We selected 10 papers among these, skipping a few with overlapping topics or where one of the authors of this paper was involved. Note that this is our classification of application papers; it might be that authors had a different opinion when they submitted their papers. We evaluated these papers for each of the seven content elements and each of the five evaluation criteria. Note that all 10 papers were accepted for the PETRI NETS conference, so it could be expected that the papers would meet expectations.

Revisiting the 10 accepted papers using our evaluation criteria resulted in the following observations:

- The scores for the five criteria were good, except for the use of tools. In particular, there were very little reflection and evaluation on the tools that had been used.
- Generality received the highest score due to the fact that relatively many papers considered a class of models for an application domain. This made the results available for the whole application domain.
- The seven content elements were more or less covered but the papers could have been improved significantly if the authors had explicitly paid attention to them.

– Calibration and validation of the model is in general a weak point. This requires systematic experimentation and for such research there is almost no tradition in computer science.
– The application of the model to the problem could be better described and the lessons learnt are difficult to find in the papers.
– In the modelling parts of the papers, little attention was paid to assumptions made and the design decisions.
– In almost all cases the evaluation according to the criteria was not obvious, since the authors did not pay explicit attention to the value of their work in terms of the evaluation criteria.

We also revisited a set of rejected papers submitted to the PETRI NETS conference in 2009. These papers were classified by the authors themselves as application paper, although we had sometimes a different opinion. We evaluated them anyway in order to see how our evaluation criteria applied to "misclassified" papers. The main observation concerning the rejected papers was that only the problem statement and the generality was meeting the expectations. So the problems were interesting enough and the potential applicability was also acceptable. A major reason was the fact that several papers were actually theory papers and theoretical results do have have wide applicability.

The above investigations show that the accepted papers satisfy our criteria (i.e. they meet or exceed expectations) whereas the rejected do not satisfy the guidelines and evaluation criteria. This, in turn, demonstrates the validity of the guidelines and associated evaluation criteria.

5 Selected Application Papers

As is evident from the discussion in section 2, application papers span a wide range of application domains and there are several criteria that makes a paper qualify as an application paper. This makes it difficult to select a single application paper to be used as a model when writing application papers. The problem is that any selection easily becomes biased towards a particular application domain or purpose. Instead, we provide references to a selected list of application papers that have been accepted for PETRI NETS in 2006–2010 [2–6]. The accepted papers are intended to be used for inspiration when preparing application papers. If using these papers as model papers, it is important to keep in mind that the papers do have weaknesses on one or more of the content elements as discussed in the previous section. The papers below encompass a wide range of application domains and they span papers dealing with a specific system to papers dealing with a class of systems.

1. D. Gilbert and M. Heiner. *From Petri Nets to Differential Equations - An Integrative Approach for Biochemical Network*. In [2], pp. 181-200.
2. Meuse N. O. Junior, S. Neto, P. Maciel, R. Lima, A. Ribeiro, R. Barreto, E. Tavares and F. Braga. *Analyzing Software Performance and Energy Consumption of Embedded Systems by Probabilistic Modeling: An Approach Based on Coloured Petri Nets*. In [2], pp. 261-281.

3. K. Winkelmann and H. Luczak. *Prospective Analysis of Cooperative Provision of Industrial Services Using Coloured Petri Nets* . In [2], pp 362-380.
4. E. Pelz and D. Tutsch. *Formal Models for Multicast Traffic in Network on Chip Architectures with Compositional High-Level Petri Nets*. In [3], pp. 381-401.
5. S. Vanit-Anunchai and J. Billington. *Modelling the Datagram Congestion Control Protocols Connection Management and Synchronization Procedures*. In [3], pp. 423-444.
6. F. Bonchi, A. Brogi, S. Corfini and F. Gadducci. *Compositional Specification of Web Services Via Behavioural Equivalence of Nets: A Case Study*. In [4], pp. 52-71.
7. R. Bouroulet, R. Devillers, H. Klaudel, E. Pelz and F. Pommereau. *Modeling and Analysis of Security Protocols Using Role Based Specifications and Petri Nets*. In [4], pp. 72-91.
8. L.G. Ding and L. Liu. *Modelling and Analysis of the INVITE Transaction of the Session Initiation Protocol Using Coloured Petri Nets*. In [4], pp. 132-151.
9. K.L. Espensen, M.K. Kjeldsen and L.M. Kristensen. *Modelling and Initial Validation of the DYMO Routing Protocol for Mobile Ad-Hoc Networks*. In [4], pp. 152-170.
10. P. Fleischer and L.M. Kristensen. *Formal Specification and Validation of Secure Connection Establishment in a Generic Access Network Scenario*. In [4], pp. 171-190.
11. G.E. Gallasch and J. Billington. *Parametric Language Analysis of the Class of Stop-and-Wait Protocols*. In [4], pp. 191-210.
12. H. Oberheid and D. Soffker. *Cooperative Arrival Management in Air Traffic Control - A Coloured Petri Net Model of Sequence Planning*. In [4], pp. 348-367.
13. I. Demongodin. *Modeling and Analysis of Transportation Networks Using Batches Petri Nets with Controllable Batch Speed*. In [5], pp. 204-222.
14. O. Gusikhin and E. Klampfl. *Integrated Process Planning and Supply Chain Configuration for Commodity Assemblies Using Petri Nets*. In [6], pp. 125-144.
15. C. Choppy, A. Dedova, S. Evangelista, S. Hong, K. Klai and L. Petrucci. *The NEO Protocol for Large-Scale Distributed Database Systems: Modelling and Initial Verification*. In [6], pp. 145-164.

Acknowledgements

The authors acknowledge the members of the Petri Nets 2010 programme committee for feedback and discussions on the guidelines, in particular the comments from Ekkart Kindler. We also acknowledge comments from Kurt Jensen (chair of the Petri Nets steering committee) on earlier versions of these guidelines.

References

1. A.R. Hevner, S. Ram, S.T. March, and J. Park. Design Sience in Information Systems Research. *MIS Quarterly*, 28(1):75–105, 2004.

2. S. Donatelli and P.S. Thiagarajan, editors. *Proc. of 27th International Conference on Application and Theory of Petri Nets and Other Models of Concurrency*, volume 4024 of *Lecture Notes in Computer Science*. Springer, 2006.
3. J. Klein and A. Yakovlev, editors. *Proc. of 28th International Conference on Application and Theory of Petri Nets and Other Models of Concurrency*, volume 4546 of *Lecture Notes in Computer Science*. Springer, 2007.
4. K. van Hee and R. Valk, editors. *Proc. of 29th International Conference on Application and Theory of Petri Nets and Other Models of Concurrency*, volume 5062 of *Lecture Notes in Computer Science*. Springer, 2008.
5. G. Franceschinis and K. Wolf, editors. *Proc. of 30th International Conference on Application and Theory of Petri Nets and Other Models of Concurrency*, volume 5606 of *Lecture Notes in Computer Science*. Springer, 2009.
6. J. Lilius and W. Penczek, editors. *Proc. of 31th International Conference on Application and Theory of Petri Nets and Other Models of Concurrency*, volume 6128 of *Lecture Notes in Computer Science*. Springer, 2010.