Stepwise Development of Distributed Vertex Colouring Algorithms (Abstract)

Manamiary Bruno Andriamiarina[1] and Dominique Méry[1]

Université Henri Poincaré Nancy 1, LORIA, BP 239, 54506 Vandœuvre-lès-Nancy, France
Manamiary.Andriamiarina@loria.fr, Dominique.Mery@loria.fr

1 Introduction

Verifying distributed algorithms is non trivial. A promising methodology is based on the Event B language and the refinement process ([1]) : it consists in redeveloping the algorithms by targeting a collection of required properties, using the refinement of models which preserves the properties of an abstract one in concrete ones. This approach has already been followed by Abrial et al for their work on the IEEE 1394 protocol ([2]). Our goal is to integrate probabilities into this approach. It appears that there are other proposals for dealing with this problem ([5, 7, 8, 4]). This paper presents an illustration of our approach through the study of vertex coloring algorithms, based on an algorithm cited by Métiver et al ([10]), and preliminary elements of a global methodology on realistic distributed algorithms.

2 The Graph Coloring Problem

Figure 1 presents our methodology. We begin by defining an abstract model, and end with three concrete models which are three different vertex coloring algorithms.

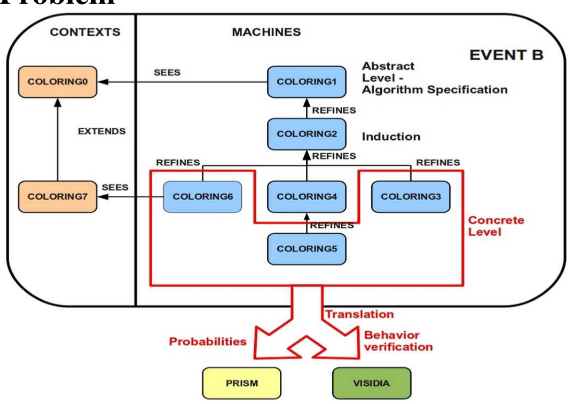

Fig. 1: Overview of the development

Specification of the graph coloring problem - This is done in the machine ***COLORING1***. A simple graph GRAPH is given over a set of vertices NODES. Using results of the graph theory, the vertices of GRAPH can be colored in such a way that no two adjacent vertices of GRAPH share the same color. The vertices of GRAPH are colored by a statement in the machine. The next step is to define an inductive coloring function which associates different colors to the adjacent vertices of GRAPH.

Computing the coloring function - In the machine ***COLORING2***, we define an inductive coloring function, which assigns the vertices their colors, which are different from the ones given to their neighbors. This model does not take into account the notion of ill-chosen colors and multiple tentatives of color choices.

The vertex coloring algorithms (COLORING3, COLORING5, COLORING6) - The algorithms defined in ***COLORING3***, ***COLORING5***, ***COLORING6*** introduce the notion that a node might make multiple color choices before finding the right one.
COLORING3 - This model presents a synchronous algorithm divided into steps : in a first round, all the vertices which have no color or are not correctly colored choose a color, then in a second round, all the vertices which have chosen colors, update the graph by deleting the edges between them and their neighbours which have chosen different colors, and finally the vertices verify if they are still linked to some neighbours; if it is the case, they restart choosing colors otherwise they stop executing the algorithm.

COLORING5 and COLORING6 - These models present asynchronous algorithms. The behavior of these algorithms are the same as the one defined in ***COLORING3***, except the fact that the vertices do not have to wait for all the others to choose their colors, update the links between them and their neighbours or restart choices.

Behavior verification - The visualisation and simulation of the algorithms are done with the software ViSiDiA ([3, 12, 11]).

Probabilistic arguments - Our analysis shows that the probabilistic aspects of these algorithms lie within the random choice of a color by a vertex. Probabilities can be distributed over of the elements of the set of available colors for a vertice or over the set of the events which are parts of the global event *"choice of a color by a node"* : the event when a node chooses a correct color happens with a certain probability, and the event when it chooses a wrong one has a certain probability attached to it, too. The next step is to analyse these aspects using probabilistic formal tools like PRISM ([6]).

3 Conclusion and Further Developments

The current stepwise development focuses on algorithms which require probabilistic arguments to achieve termination. We have derived these vertex coloring algorithms from an abstract specification of the graph coloring problem. They have in common the integration of possible errors during the choice of colors by vertices and this is the point where probabilistic arguments can be handled. As we have mentioned in the introduction, this paper presents preliminary elements of a global methodology for handling the correct-by-construction refinement-based approach applied to distributed algorithms and further work is needed to adapt technique and tools. It is also a first step towards our goal of obtaining a development framework, integrating probabilistic arguments and refinement, for distributed algorithms. This framework can be an extension of the Event B method ([5, 7]) or, in our case, a combination of the Event B method and other formal methods (see figure Fig.1).

References

1. Jean-Raymond Abrial. *Modeling in Event-B: System and Software Engineering*. Cambridge University Press, first edition, June 2010.
2. Jean-Raymond Abrial, Dominique Cansell, and Dominique Méry. A mechanically proved and incremental development of ieee 1394 tree identify protocol. *Formal Asp. Comput.*, 14(3):215–227, 2003.
3. Bilel Derbel. A brief introduction to visidia. `http:www.lifl.fr/~derbel/resources_visidia/visidia_tutorial.pdf`, 2007.
4. K. R. Duffy, N. O'Connell, and A. Sapozhnikov. Complexity analysis of a decentralised graph colouring algorithm. *Inf. Process. Lett.*, 107:60–63, July 2008.
5. Stefan Hallerstede and Thai Son Hoang. Qualitative probabilistic modelling in event-b. In *IFM*, pages 293–312, 2007.
6. M. Kwiatkowska, G. Norman, and D. Parker. PRISM: Probabilistic symbolic model checker. In T. Field, P. Harrison, J. Bradley, and U. Harder, editors, *Proc. 12th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation (TOOLS'02)*, volume 2324 of *LNCS*, pages 200–204. Springer, 2002.
7. Carroll Morgan, Thai Son Hoang, and Jean-Raymond Abrial. The challenge of probabilistic *event b* - extended abstract. In *ZB*, pages 162–171, 2005.
8. Carroll Morgan, Annabelle McIver, and Karen Seidel. Probabilistic predicate transformers. *ACM Trans. Program. Lang. Syst.*, 18(3):325–353, 1996.
9. Dominique Méry. Refinement-based guidelines for constructing algorithms. In Jean-Raymond Abrial, Michael Butler, Rajev Joshi, Elena Troubitsyna, and Jim C. P. Woodcock, editors, *Refinement Based Methods for the Construction of Dependable Systems*, number 09381 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany, 2010. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany.
10. Y. Métivier, J.M. Robson, N. Saheb-Djahromi, and A. Zemmari. An analysis of an optimal bit complexity randomised distributed vertex colouring algorithm (extended abstract). *OPODIS*, pages 359–364, 2009.
11. Afif Sellami. *Des calculs locaux aux algorithmes distribués*. PhD thesis, Université Bordeaux I, Talence, France, 2004.
12. Mohamed Tounsi, Ahmed Hadj-Kacem, Mohamed Mosbah, and Dominique Méry. A Refinement Approach for Proving Distributed Algorithms : Examples of Spanning Tree Problems. In *Integration of Model based Formal Methods and Tools(IMFMT 2009)*, Düsseldorf Allemagne, 2009.