Proving Properties of the Itai Rodeh Leader Election Protocol for Any Number of Processes

D.Graham* M.Calder†

Abstract

Verification of randomised distributed protocols can be carried out using probabilistic model checking with tools such as *PRISM*. However, model checking is limited in that it can only verify systems with a fixed number of processes. The problem of verifying the family of systems of size N, for any N, is known as the parameterised model checking problem and has been shown to be undecidable. However, some work has been done to tackle the problem for certain classes of system.

In [1] Miller & Calder describe an approach for the IEEE 1394 Firewire leader election protocol, which they model non-probabilistically using the SPIN model checker. This method is based on the protocol being *degenerative*, a term introduced in [1] to denote any system that, for any given size, will eventually behave as a smaller version of the system. This property allowed them to employ an inductive proof on the size of the system in order to demonstrate that certain properties hold regardless of the number of processes.

Our first result is to adapt the proof given by Miller & Calder for a probabilistic version of the Firewire leader election protocol (modelled in PRISM). We then turn our attention to verifying certain probabilistic properties of the Itai Rodeh leader election protocol for rings, (using PRISM to model the protocol). This protocol is also degenerative so we outline how the technique employed for the Firewire leader election protocol could be applied.

1 Introduction

Model checking is a verification method whereby a model of a system can be defined and its properties specified and then proven to be correct, or otherwise, by an automated search of the complete state space of the model. Model checking is particularly suited to verification of systems of concurrently executing processes but is restricted to verifying systems with a fixed number of processes. Proving a property for a system with N identical processes, for any N is known as the *parameterised model checking problem*. This problem is undecidable in general but techniques can be used to solve it for certain types of system. For example, a common method employed for certain systems is the construction

*D.Graham, Dept. of Computing, University of Glasgow , UK. `Email:doug@dcs.gla.ac.uk`

†M.Calder, Dept. of Computing, University of Glasgow , UK. `Email:muffy@dcs.gla.ac.uk`

of a *network invariant.* If such a construct exists, then any properties that are verified for the invariant will also hold for any particular size of the system under consideration.

Probabilistic model checking augments traditional model checking by allowing probabilistic choice to be introduced into the definition of a system's model. Probabilistic model checking has become an important area of research due to the increased use of probabilistic algorithms and the requirement for analysis of not just system correctness but also system performance. Probabilistic model checkers, such as PRISM, enable properties such as *"the system will fail with probability less than 0.01"* and *"with probability 1, the system will terminate within 100 time units"* to be specified.

As for classical model checking, probabilistic model checking is restricted to verifying systems of a fixed size. Again, certain classes of system have been verified for an arbitrary number of processes but in general, the parameterised model checking problem for probabilistic systems has not been widely addressed.

2 Parameterised model checking of the Firewire protocol

In [1], Miller and Calder describe a method for tackling the parameterised model checking problem for the IEEE 1394 Firewire leader election protocol, which they model non-probabilistically and verify using the SPIN model checker.

The Firewire leader election protocol is designed to elect a leader from a set of processes arranged in an acyclic topology. Each process may send one of three messages to a neighbouring process: *be_my_parent, be_my_child* or *acknowledge.* Any process that has received *be_my_parent* messages from all or all but one of its neighbours responds to these requests with *be_my_child* messages and, if necessary, sends a *be_my_parent* to the remaining neighbour. The neighbouring processes will send an *acknowledge* upon receiving a *be_my_child.* In this manner the protocol essentially builds a spanning tree with the root process elected as leader.

After sending an *acknowledge* message a process no longer participates in the protocol. Calder and Miller term this behaviour *degenerative* and demonstrate that a system of any given size will eventually *degenerate* and behave like a smaller system. Thereby, they are able to prove, by induction on the size of the system, that any property that holds of a model of the 'base' system will hold for a a model of any system configuration. Essentially the proof relies on showing that any behaviour or path in a model of the system of a given size is equivalent (ignoring the variables of the 'degenerated' processes) to a path in a model of a smaller system, and vice versa. Thus a relationship is established between any system model and a set of models of smaller systems. Thus, by induction the proof goes through.

Note that, although the Firewire leader election protocol is modelled as a non-probabilistic system by Miller and Calder using SPIN, the protocol is probabilistic. Specifically, it is possible that two processes will end up in a contention state whereby they both attempt to be elected leader. In order to resolve this, a probabilistic choice is made over timing delays, so that both processes will "back-off" for a certain time before attempting to be elected leader again. By modelling two processes in a contention situation using PRISM, Kwiatkowska et al. [2] have shown that, with probability 1, any contention will eventually be resolved (i.e. a leader will be elected). Using this result, Miller & Calder were able to represent the resolution of leader contention abstractly as the result of a non-deterministic choice between the contending processes.

However, it is possible to extend their model so that the contention is resolved probabilistically. More precisely, we can model the contention resolution with both contending processes making a simple probabilistic coin flip. If both processes choose the same value then they flip again, otherwise, one process is elected leader.

We can model the protocol in this way and verify it for fixed configurations using PRISM. We can then extend the proof of Miller & Calder in a relatively straightforward manner in order to show that probabilistic variants of the properties that held for the non-probabilistic model still hold in the probabilistic one.

3 The Itai Rodeh leader election protocol

Having successfully employed this inductive proof approach for the Firewire protocol, we want to determine other systems for which it may be applicable. Therefore, we consider a particular randomised distributed protocol designed to elect a leader from a set of processes arranged in a ring: explicitly the asynchronous version of the Itai Rodeh leader election protocol [3].

The algorithm proceeds in two phases. Initially all processes are deemed *active*. In the first phase each process selects 0 or 1, each with probability a half. Every process then sends its selection to the next process in the ring (the ring being unidirectional). When a process receives its neighbours choice, if it is 1 and its own choice was 0 then the process will become *passive*. For any other combination of choices the process remains active and then transmits a counter which is passed around the ring. Each time this counter passes through a passive process it is incremented by one. Therefore, if the active process receives a counter with value $N-1$ (where N is the size of the ring) it knows that it is the only remaining active process and so can declare itself leader.

The Itai Rodeh protocol is degenerative because once a process becomes passive it will only pass on messages it receives. However, note that the maximum value of the counter is dependent on the number of processes. This dependency means that the system is not fully degenerative as for the Firewire protocol.

4 Parameterised model checking of the Itai Rodeh leader election protocol

The Itai Rodeh protocol has been modelled and verified (for rings of size less than ten) using the PRISM probabilistic model checker (see [4, 5]). We have adapted the PRISM definition given on [4] with asynchronous rather than synchronous communication so that neighbouring processes transmit along a channel of length N, where N is the size of the ring. Let $\mathcal{M}_N$ denote the underlying model (Markov Decision Process) for our PRISM definition for a ring of size N. Then we aim to prove the following.

Conjecture 1 *For any size of ring, $N \geq 3$, the model $\mathcal{M}_N$ satisfies any PCTL property ϕ that does not contain the next-time or time-bounded until operator and does not refer to any process' local variables.*

We now give an informal description of an (incomplete) proof of Conjecture 1.

In order to prove the above theorem we want to show that there is some relationship between $\mathcal{M}_N$ and $\mathcal{M}_{N+1}$. In the case of the probabilistic Firewire model we were able to show that each 'behaviour' (i.e. each DTMC obtained under some adversary) of a model was weakly bisimilar to the behaviour of some smaller model and vice versa.

However, it is important to note that, in the case of the probabilistic version of Firewire mentioned above, the next set of processes to be 'removed' are chosen non-deterministically whereas for the Itai Rodeh protocol the next process to become passive is chosen probabilistically. In addition to this, when a set of processes are removed in the Firewire protocol, they are no longer involved in the protocol, whereas a passive process in the Itai Rodeh protocol still participates by passing on messages. Moreover, the counter that is passed around the ring in the Itai Rodeh protocol is dependent on the size of the ring whereas none of the variables in the Firewire depend on the number of processes in the system.

To deal with these issues we introduce a series of intermediate models. We first deal with the size of the channels, which is dependent on N. We do so by creating a definition that is identical to the definition of $\mathcal{M}_N$ but that has channels with a capacity of $N+1$. From Lemma 4 it should be apparent that the model given by this definition, which we denote $\mathcal{M}_N^{c+1}$, is isomorphic to $\mathcal{M}_N$.

Lemma 1 *For $\mathcal{M}_N$, the maximum number of messages present in the channels at any one time is N.*

We then then produce a definition of a system of size $N+1$ that initially makes a non-deterministic choice over the set of processes, setting one of the processes to be passive immediately. This passive process behaves as it would do normally except that it does not increment the counter when it receives it, so the counter will still be of maximum size N. Apart from this, the definition is as for the definition for the model $\mathcal{M}_{N+1}$. We denote this new model by $\mathcal{M}_N^P$ and can show that:

Lemma 2 *For $\mathcal{M}_N^P$ the maximum number of messages ever present in the channels at any one time is N.*

Lemma 3 *If process P is the first process to become passive in $\mathcal{M}_N^P$ then the maximum number of messages contained in the channels between process $P-1$ and $P+1$ is N (where addition and subtraction is modulo N). Note that if process P has received a message and has not yet sent it on then this is included in the count of messages.*

If we first assume that process $N+1$ is the first process to become passive then by the above two lemmas we can define a surjective map from any state in $\mathcal{M}_N^P$ for which one or more messages is present in the channels between process N and process 1 to a state in $\mathcal{M}_N^{c+1}$ for which the equivalent message(s) is (are) present in the buffer between process N and process 1.

Under this mapping we can show that for each behaviour of $\mathcal{M}_N^{c+1}$ there exists a weakly bisimilar behaviour of $\mathcal{M}_N^{c+1}$, and vice versa. Weak bisimulation for countable state DTMCs is defined in [6] (under weak bisimulation next-time and time-bounded until PCTL properties are not preserved between models). By symmetry it is obvious that this is the case if any of the other processes is the first to become passive. Note that this means, however, that we cannot verify properties that refer to the local variables of a single process.

Next we define a model that is as for $\mathcal{M}_N^P$ but the non-deterministically chosen passive process now increments the counter whenever it receives it. We denote this model $\mathcal{M}_{N+1}^P$. If we assume that P is the first process to become passive then we can again define a surjective mapping between the states of $\mathcal{M}_N^P$ and $\mathcal{M}_{N+1}^P$. Specifically, consider any state of $\mathcal{M}_{N+1}^P$ for which some passive process, Q say, has a counter value of $k > 0$. If the distance (measured in number of processes) in an anti-clockwise direction between Q and P is less than k then we can map this state to an equivalent state in $\mathcal{M}_N^P$ where Q's counter has value $k-1$. Again, under this mapping, for each behaviour of $\mathcal{M}_N^P$ we can show that there is an equivalent behaviour in $\mathcal{M}_{N+1}^P$ and vice versa, where equivalence is described by strong bisimulation (as defined in [6]).

Finally, we want to show that $\mathcal{M}_{N+1}^P$ is related to $\mathcal{M}_{N+1}$. The difficulty here is that in $\mathcal{M}_{N+1}$ the first passive process is chosen probabilistically whereas in $\mathcal{M}_{N+1}^P$ it is chosen non-deterministically. We conjecture that the two models are related by weak bisimulation but the resolution of this problem is work in progress. If we assume that this holds then we have established that Conjecture 1 is true.

5 Conclusion & Future Work

In this paper we have described an inductive proof technique for a class of randomised distributed algorithms denoted as *degenerative*. We have applied this method to two systems, firstly extending the proof given for a non-probabilistic version of the IEEE 1394 Firewire protocol to a probabilistic version and secondly applying the technique to the Itai Rodeh leader election protocol for rings. Using this approach we have shown that any PCTL property that does not refer to the local variables of a process will hold for the Firewire protocol for an arbitrary number of processes and we conjecture that this is true of the Itai Rodeh protocol.

With respect to future work, our initial goal is to complete the proof of Conjecture 1. Our subsequent goal is to consider other randomised distributed algorithms that may be degenerative. One possibility is the weak shared coin protocol of Aspnes & Herlihy [7]. This protocol uses a shared memory counter which can be incremented or decremented by any process according to the result of a coin flip. When the counter goes above (below) a certain threshold value after a process has updated it then that process will choose heads (tails) and terminates. Therefore the protocol degenerates but as the size of the shared counter is dependent on N, the protocol is partially degenerative in the same sense as the Itai Rodeh leader election protocol. Ultimately we would then like to establish a standardised induction proof technique that could be used for any system that could formally be defined as degenerative. This would require establishing a set of rules in the definition of a model that would guarantee that the induction proof could be used.

References

[1] A. Miller and M. Calder, "Two approaches for verifying the tree identify phase of the ieee 1394 (firewire) protocol for any size of network," Tech. Rep. TR2003-144, University of Glasgow, Department of Computing Science, 2003.

[2] C. Daws, M. Kwiatkowska, and G. Norman, "Automatic verification of the IEEE 1394 root contention protocol with KRONOS and PRISM," in *Proc. 7th International Workshop on Formal Methods for Industrial Critical Systems (FMICS'02)* (R. Cleaveland and H. Garavel, eds.), vol. 66.2 of *Electronic Notes in Theoretical Computer Science*, Elsevier, 2002.

[3] A. Itai and M. Rodeh, "Symmetry breaking in distributed networks," *Information and Computation*, vol. 88, no. 1, pp. 60–87, 1990.

[4] "PRISM home page." http://www.cs.bham.ac.uk/ dxp/prism/.

[5] W. Fokkink and J. Pang, "Simplifying Itai-Rodeh leader election for anonymous rings," in *Proc. 4th Workshop on Automated Verification of Critical Systems (AVoCS'04)*, vol. 128(6) of *Electronic Notes in Theoretical Computer Science*, pp. 53–68, Elsevier Science, 2004.

[6] C. Baier, H. Hermanns, J.-P. Katoen, and V. Wolf, "Comparative branching-time semantics for Markov chains," *Information and Computation*, vol. 200, pp. 149–214, Aug 2005.

[7] J. Aspnes and M. Herlihy, "Fast randomized consensus using shared memory," *Journal of Algorithms*, vol. 11, pp. 441–461, Sep 1990.