THE COMPLEXITY OF COLLAPSING REACHABILITY GRAPHS

Sudhir Aggarwal

Bell Communications Research
Morristown, N.J. 07960

Daniel Barbara

C.S. Department, Princeton University
Princeton, New Jersey 08540

Walter Cunto

Centro Cientifico IBM
A.P. 388, Caracas, Venezuela

Michael R. Garey

AT&T Bell Laboratories
Murray Hill, New Jersey 07974

1. Introduction

There is an increasing proliferation of communication protocols, ranging from low level physical layer to the application layers. Consequently, more and more tools are being developed for specification and validation of protocols. Many of these tools are based on describing the protocols as a set finite state machines ([BM80],[RW83],[ABM88],[HK89].) Validation of such protocols is generally based on the following approach. The global states of the protocol reachable from an initial state are determined using the descriptions of the component machines that make up the protocol and the rules for their composition. The result of this procedure defines a graph, commonly referred to as the *reachability graph*. This directed graph has vertices that correspond to global states of the protocol, and edges that represent possible state transitions. Thus, paths in the directed graph describe possible state trajectories and can be used to answer questions about the dynamic behavior of the protocol. For realistic protocols, the size of the reachability graph may be extremely large, making its analysis difficult. Thus, some reduction techniques must be used to make the analysis possible.

When the protocol is composed of many processes, the designer would often like to focus on a particular subset for analysis purposes. Consider, for instance, a protocol consisting of a

transmitter process, a receiver process, and processes describing the communication facilities. The designer might be interested in looking only at the joint moves of the transmitter and receiver processes, abstracted from the rest of the involved processes. If this is the case, some of the states in the reachability graph become equivalent, and can be merged into a single state, thereby making the analysis of the protocol more manageable.

Reduction techniques must be based on preserving certain properties of the original graph that are of interest to the designer. A minimal property that one might wish to preserve is making sure that a path in the collapsed graph implies that an equivalent graph exists in the original graph.

Consider the following procedure of defining a smaller, collapsed graph from the original reachability graph. Let the protocol consist of k processes. Process i has state space $V^i = \{v_{1i}, v_{2i}, ..., v_{n_i i}\}$, where every v_{ji} denotes a local state of process i. A state of the reachability graph is a k-tuple $s = \langle s_1, s_2, ..., s_k \rangle$ where $s_i \in V^i$. We call s a global state. Let us consider a particular subset of processes whose behavior is of interest, $I = \{i_1, i_2, i_3, ..., i_j\}$ for some $j \leq k$. Focusing on I means making a projection of each global state $s = \langle s_1, s_2, ..., s_k \rangle$ in the reachability graph onto the elements in I, i.e., a projected state $\pi_I(s) = \{s_{i_1}, s_{i_2}, ..., s_{i_j}\}$. Two global states s' and s'' become equivalent if their projected states onto I are equal, that is, $\pi_I(s') = \pi_I(s'')$.

By specifying I, the designer also specifies a set of equivalence classes, each one consisting of those global states that are equivalent under the projection.

In addition to specifying the set of equivalence classes for the vertices of the graph, we must also specify the new set of edges. The natural way to do this is to include an edge in the collapsed graph between two nodes $\pi_I(s)$ and $\pi_I(s')$ if there exist an edge from an element of the preimage $\pi_I^{-1}(s)$ to an element of $\pi_I^{-1}(s')$. The set of edges along with the set of projected states defines the collapsed graph.

Such a collapsing decreases the size of the graph, but does not always preserve enough information for the analysis. It is not enough to define equivalence classes of states and proceed

to collapse them. Such an arbitrary procedure may lead to misleading information. For example, consider the graph of Figure 1.1.a.

If we consider a projection onto the processes $I = \{1, 2, 3\}$, the equivalence classes are as follows:

$$C = \{\{A\}, \{B\}, \{C\}, \{D, E\}, \{F\}, \{G\}\}.$$

Collapsing nodes within the same equivalence class leads to the graph of Figure 1.3.b.

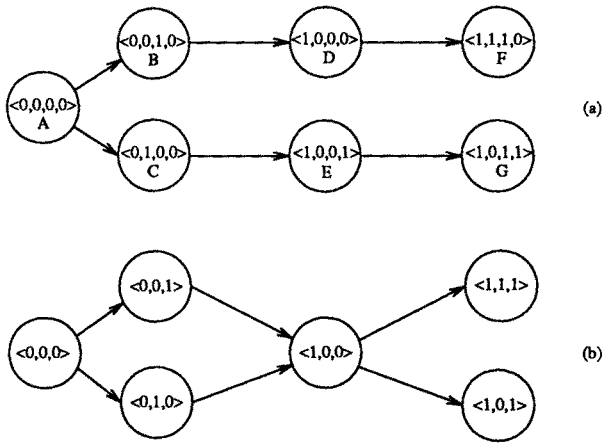

Figure 1.1 A misleading collapsing.

However, in this graph we would infer that there is a way of getting from the macro-state $\langle 0, 0, 1 \rangle$ to the macro-state $\langle 1, 0, 1 \rangle$ whereas this is not the case in the original graph. That is, in the original graph, we cannot get from any state that collapses to $\langle 0, 0, 1 \rangle$ to any state that collapses to $\langle 1, 0, 1 \rangle$.

This change in the structure of the graph may lead the designer to erroneous conclusions. Consequently, given a reachability graph and the projection, we still have to find the minimum partition of states within the same equivalence class that can be safely collapsed, i.e., without creating artificial paths. Unfortunately, as we prove later on, the problem of finding the largest subsets that can be collapsed safely is a very difficult one for which no efficient algorithm is likely.

The notion of protocol projection has been used before by Bochmann and Merlin [BM80] in the context of describing an operation for protocol synthesis. Lam and Shankar [LS84] describe a method of stepwise refinement to transform a protocol in to a series of smaller, single-function protocols called by them image protocols. However, their approach of collapsing differs from ours in the fact that a particular assertion about the protocol is used as the termination condition for the algorithms. In our approach, the collapsing is made based only in the graph properties and can be used subsequently to analyze the protocol under different assertions. Moreover, it is not clear in [LS84] what the complexity of the projecting algorithms is. In the next section we introduce two definitions of path preserving collapsing, which we shall call *weak* and *strong* path preserving collapsing. Although the first type of collapsing does not preserve all the properties of the original graph, it is still a useful tool for analysis. Moreover, since the weak collapsing is less restrictive, it is conceivable that the reduction in the size of the protocol achieved by this type of collapsing will be significantly larger than the one obtained by applying the strong collapsing. It can be shown that the conditions used in [LS84] to preserve liveness and safety assertions, imply that the image protocol is found by a strong path preserving collapsing [ABCG88]. We show in this paper that finding optimal collapsings under both definitions is an NP-Complete problem, and thus it is very unlikely that any efficient method exists. The equivalence of the method in [LS84] to strong path preserving collapsing implies that this complexity result applies to their method as well. That is, an efficient method of finding the smallest image protocol is very unlikely to exist.

This paper analyzes the complexity of safely collapsing a reachability graph based on a protocol specification. We first give a formal statement of the problem in section 2, stating the two definitions for path preserving collapsing. In section 3 we show that this problem is NP-complete. Finally, some ideas about further research are given in section 4.

2. Characterization of Collapsing

As discussed in the introduction, in order to insure the preservation of graph properties of interest, some restrictions on the collapsing must be imposed. The following two definitions state

precisely the properties we are dealing with in this paper.

Definition 2.1. A collapsing is *weak path preserving* if the following condition holds. There is a path from node A to node B in the collapsed graph if and only if there exists a path in the original graph from some node a to some node b such that a was collapsed to A and b was collapsed to B.

□

Definition 2.2. A collapsing is *strong path preserving* if the following condition holds. There exists a path from node A to node B in the collapsed graph if and only if there exists a path in the original graph from **every** node a to **every** node b such that a was collapsed to A and b was collapsed to B. □

Preserving paths weakly allows us to answer questions about paths in the collapsed graph, being assured about their validity in the original reachability graph. Using that transformation, we do not concern ourselves with other types of questions, for instance questions posed as general temporal logic expressions, for which it is also crucial to preserve information such as the possibility of infinitely looping among the nodes belonging to the same equivalence class.

Stronger restrictions are imposed over the collapsing by the strong path preserving property allowing, as we shall show, these questions to be answered in the collapsed graph. However, it is very likely that imposing stronger restrictions minimizes the gains of size reduction in the collapsing graph. In what follows, we use the term path preserving to refer to any of the two definitions indistinctly. We shall prefix the term with the words "weak" or "strong" to refer to an specific type of path preserving collapsing.

Definition 2.3. Let $G=(V, E)$ be a simple directed graph and let Π be a partition of V. (A partition of V is a set of mutually disjoint subsets of V whose union is V.) Let V' denote a set of nodes whose elements are in one-to-one correspondence with the subsets of Π, and let $\Pi(a)$ denote the node in V' that corresponds to the subset of Π containing node $a \in V$. We say that $G'=(V', E')$ is the *collapsed graph* of G with respect to Π if, for any $A, B \in V'$, the edge (A, B) belongs to E' if and only if there exist nodes $a, b \in V$ with $\Pi(a)=A$ and $\Pi(b)=B$ such that the

edge (a, b) belongs to E.

Let $COLLAPSE(G, \Pi)$ denote the collapsed graph of G with respect to Π. It includes an edge joining two macro-node if and only if two nodes of G, one belonging to each macro-node, are joined by such an edge (in the proper direction). Hence it is easy to see that $COLLAPSE(G, \Pi)$ can be constructed from G and Π in time proportional to the number of edges in G.

The graph $COLLAPSE(G, \Pi)$ is not necessarily path preserving. There is, however, a nice characterization of a weak path preserving collapse. Let $CLOSURE(G)$ denote the transitive closure of G, which has an edge from a to b if and only if G contains a path from a to b. (See [AHU74], section 5.7, for details of how to compute transitive closure in polynomial time.) Then, as we shall prove, G' is a weak path preserving collapsed graph of G (with respect to Π) if and only if

$$CLOSURE(COLLAPSE(G, \Pi)) = COLLAPSE(CLOSURE(G), \Pi)$$

whenever this latter condition holds, we shall say that the collapsing is commutative. Figure 2.1 shows two examples of collapsing; the first is path preserving and the second is not.

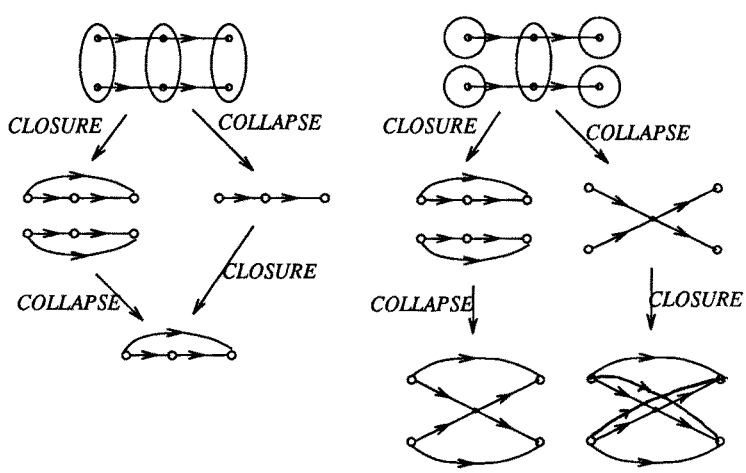

Figure 2.1

We now prove the equivalence of the notions of weak path preserving and the commutativity of the *CLOSURE* and *COLLAPSE* functions.

Theorem 2.1. The graph $G' = COLLAPSE(G, \Pi)$ is weak path preserving if and only if

$$CLOSURE(COLLAPSE(G, \Pi)) = COLLAPSE(CLOSURE(G), \Pi).$$

Proof: An edge (A, B) belongs to $CLOSURE(G')$ if and only if there exists a path in G' from node A to node B. An edge (A, B) belongs to $COLLAPSE(CLOSURE(G), \Pi)$ if and only if there exist nodes $a, b \in V$ with $\Pi(a)=A$ and $\Pi(b)=B$ such that (a, b) is an edge of $CLOSURE(G)$. The latter holds if and only if there is a path from a to b in G. Hence the condition that the operations of *CLOSURE* and *COLLAPSE* commute for G and Π is equivalent to saying that there exists a path in G' from A to B if and only if there exist nodes $a, b \in V$ with $\Pi(a)=A$ and $\Pi(b)=B$ such that G contains a path from a to b. But this is equivalent to the definition of a weak path preserving collapse, from which the theorem follows. □

Unfortunately, this characterization is not valid for strong path preserving collapsed graphs, as we can easily see in the first graph of Figure 2.1. There, the collapsing is not strong path preserving and yet the two operations are commutative.

As we have shown in Figure 2.1, not every collapsed graph is path preserving. When G' is not a path preserving collapsed graph of G with respect to Π, the problem will be to find a refined partition Π' of Π such that $COLLAPSE(G, \Pi')$ is path preserving. Note that the problem can always be solved since the partition that places each vertex of G in a singleton subset by itself gives a valid, though trivial, solution.

A *refined partition* Π' of a partition Π is formally defined to be any partition in which each element (subset) is contained in some element of Π. Any refined partition has cardinality at least as large as that of the original partition.

In the task of finding a refined partition, it is desirable to look for one that is as similar as possible to the initial partition. Thus, we can state the following optimization problem: Given a directed graph $G=(V, E)$ and a partition Π of V, find a refined partition Π' of Π such that

COLLAPSE(G, Π') is path preserving and the cardinality of Π' is as small as possible.

3. Complexity Result

In this section, we focus on proving that finding an optimal refined partition is an NP-Complete problem.

The corresponding decision problem is formulated as follows.

PATH PRESERVING REFINED PARTITION (PPRP)

INSTANCE:

A directed graph $G = (V, E)$, a partition Π of V, and an integer B such that $|\Pi| \leq B \leq |V|$,

QUESTION:

Is there a refined partition Π' of Π with $|\Pi'| \leq B$ such that *COLLAPSE*(G, Π') is path preserving?

Note that the definition of **PPRP** has been given in general terms and applies to both types of path preserving collapsing. We shall use this definition to prove that in both instances, the problem is NP-Complete.

In order to prove that the decision problem is in the family of NP-Complete problems, we will follow the approach used in [GJ79].

Lemma 3.1. *PPRP* $\in$ *NP*.

Proof: PPRP is easily seen to be in NP. A nondeterministic algorithm for it need only guess a refined partition Π' of Π and check if Π' has the proper cardinality and if *COLLAPSE*(G, Π') is path preserving. The guessing step needs only nondeterministic polynomial time. Following the discussion presented above, the checking step can be computed in deterministic polynomial time. Thus, the first requirement for NP-Completeness is met. □

For the second requirement, GRAPH K-COLORABILITY, known to belong to the family of NP-Complete problems [GJ79, pp. 191] is chosen as the problem to reduce to PPRP. The problem is as follows:

INSTANCE: Undirected graph $G = (V, E)$, positive integer $K \leq |V|$.

QUESTION: Is G K-colorable, i.e., does there exist a function $f : V \rightarrow \{1, 2, ..., K\}$ such that $f(u) \neq f(v)$ whenever $\{u, v\} \in E$?

The transformation function that converts an instance of GRAPH K-COLORABILITY to an instance of PPRP is fairly straightforward. Let $\hat{G} = (\hat{V}, \hat{E})$ be the undirected graph with $V = \{\hat{v}_1, ..., \hat{v}_n\}$. We transform this to a *directed* graph $G = (V, E)$ where for each vertex $\hat{v}_i$ in $\hat{V}$, we create a set of three vertices v_i, v_i', v_i'' in V with *direct* edges from v_i to v_i' and v_i' to v_i''. There are $3n$ vertices in V, and $2n$ such direct edges. See Fig. 3.1.

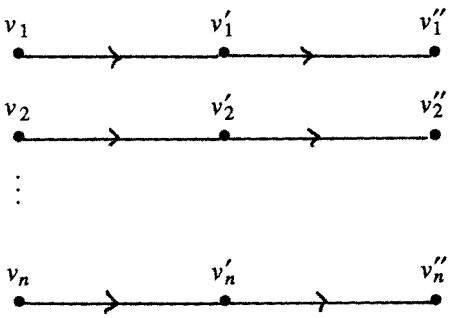

Figure 3.1 Vertices and direct edges in G.

We next add a set of *cross* edges as follows. For each edge that is *not* in $\hat{E}$, i.e., $\{\hat{v}_i, \hat{v}_j\} \notin \hat{E}$, we add the cross edges (v_i, v_j'), (v_j, v_i'), (v_i', v_j'') and (v_j', v_i''). There are four such cross edges added for each edge not in $\hat{E}$. See Figure 3.2.

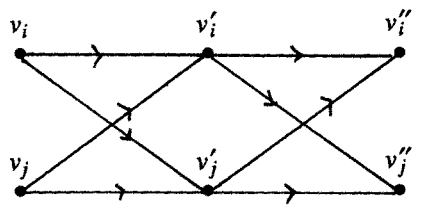

Figure 3.2 Adding cross edges.

We now consider the partition Π in which v_1', v_2', ..., v_n' belongs to the same subset and all other vertices belong to singleton subsets. Thus $|\Pi| = 2n + 1$. If we let $B = 2n + K$, the

transformation is completely defined. Notice that if two vertices in $v'_1, v'_2, ..., v'_n$ are collapsed, the transformation is both weak and strong path preserving.

Suppose the original graph $(\hat{G}, \hat{E})$ is K-colorable. Then the vertices corresponding to each color form an independent set; that is, for any u, w with $f(u)=f(w)$, there does not exist an edge $\{u, w\}$ in $\hat{E}$. Consequently, the vertices in $\{v'_1, ..., v'_n\}$ corresponding to this color can safely be collapsed without introducing any new paths. There will be at most K such macro-nodes in addition to the $2n$ singleton nodes, and thus a refined path preserving partition exists of size at most $B = 2n + K$.

Conversely, suppose a refined partition of V' of size at most $B = 2n + K$ is possible. Note that any vertex in the sets $\{v_1, ..., v_n\}$ and $\{v''_1, ..., v''_n\}$ cannot be in a partition class with any other vertex, because they are all in singleton sets at Π. Thus, only vertices in $\{v'_1, ..., v'_n\}$ can form macro-nodes. Since $2n$ vertices must be singleton, there are at most K such macro-nodes. Now, collapsing the vertices onto the macro-nodes does not generate any new paths, so it must be that the vertices in each macro-node form an independent set. Consequently, the graph $(\hat{G}, \hat{E})$ is K-colorable.

It is easy to see that the transformation that converts an instance of GRAPH K-COLORABILITY to an instance of PPRP is a polynomial transformation. We have arrived at the main result of this section.

Theorem 3.1. PPRP belongs to the family of NP-Complete problems.

Proof: By the arguments above. □

4. Conclusions

We have presented the idea of collapsing reachability graphs to obtain more manageable graphs that preserve certain properties and are easier to analyze. We showed that the problem of finding an optimal path preserving collapse is NP-Complete under two definitions of collapsing while preserving path properties. In [ABCG88] we also show that even an approximation algorithm for finding a "good" path preserving collapsed graph, with number of nodes

guaranteed to be within a factor of 2 of the minimum number of nodes, is unlikely to exist. In that paper we also discuss sufficiency conditions for the graph that make the collapsing always path preserving.

Although the main result of this paper is discouraging, it must be kept in mind that in many cases the graphs will satisfy stronger conditions that may allow obtaining the optimal collapsed graph. Furthermore, heuristic techniques can produce useful collapsed graphs even when they may not correspond to the optimal solution.

REFERENCES

[ABCG88] S. Aggarwal, D. Barbara, W. Cunto and M. Garey, "Collapsing Reachability Graphs," submitted for publication.

[ABM88] S. Aggarwal, D. Barbara, K. Z. Meth, "A Software Environment for the Specification and Analysis of Problems of Coordination and Concurrency," *IEEE Transactions on Software Engineering*,Vol. 14-3, March 1988.

[AHU74] A. Aho, J. Hopcroft, and J. Ullman, *The Design and Analysis of Computer Algorithms*, Addison-Wesley, 1974.

[BM80] G. V. Bochmann and P. Merlin, "On the Construction of Communication Protocols," in *Proceedings of the 5th ICCC,* Atlanta, Oct. 1980.

[GJ79] M. R. Garey and D. S. Johnson, *Computers and Intractability*, W. H. Freeman, 1979.

[HK89] Z. Har'El and R. Kurshan, "COSPAN: A Software System for Analysis of Coordination, " this issue.

[LS84] S. Lam and U. Shankar, "Protocol Verification via Projections," *IEEE Transactions on Software Engineering, Vol.10, No. 4*, July 1984.

[RW83] H. Rudin and C. H. West, (eds.), *Protocol Specification, Testing, and Verification, III*, North-Holland, 1983.