The maximum number of states after projection

THE MAXIMUM NUMBER OF STATES AFTER PROJECTION

BY

HUUB M.J.L. SCHOLS

87/08

April 1987

COMPUTING SCIENCE NOTES

This is a series of notes of the Computing Science Section of the Department of Mathematics and Computing Science of Eindhoven University of Technology.
Since many of these notes are preliminary versions or may be published elsewhere, they have a limited distribution only and are not for review.
Copies of these notes are available from the author or the editor.

Eindhoven University of Technology
Department of Mathematics and Computing Science
P.O. Box 513
5600 MB EINDHOVEN
The Netherlands

editor: F.A.J. van Neerven

The maximum number of states after projection

Huub M.J.L. Schols

Department of Mathematics and Computing Science
Eindhoven University of Technology
Eindhoven, the Netherlands

ABSTRACT

Projecting a minimal deterministic state graph onto an alphabet might yield a minimal deterministic state graph that contains more states than the original one, due to the introduction of nondeterminism, cf. [Kaldewaij0, example 5.4, p. 36]. In this paper we show that for every natural number N , $N \geq 2$, there exists an alphabet A and a minimal deterministic state graph S, that contains exactly N states, such that projecting S onto A yields a minimal deterministic state graph that contains $(3 * 2^{(N-2)} - 1)$ states. It is easily shown that $(3 * 2^{(N-2)} - 1)$ is the upper limit. Robert Huis in 't Veld was the first who showed us this upper limit.

0 Introduction

This problem arises from studying communicating processes, cf. [Hoare] and [Kaldewaij0]. Transitions in a state graph denote communication actions in which a process may involve. Projecting transitions away corresponds to hiding communication actions. These are referred to as internal moves or ϵ-transitions. In the remainder of this paper an alphabet is a set of symbols, that denote transitions. Furthermore, by referring to *state graph* we mean *minimal deterministic state graph*. $x \xrightarrow{a} y$ denotes a transition labeled a from the state labeled x to the state labeled y.

1 Projecting a state graph onto an alphabet

Consider a state graph S with N , $N \geq 2$, states labeled with natural numbers from 0 up to $(N-1)$. Projecting S yields a state graph, say T . Due to introduction of nondeterminism states of T correspond to subsets of states of S , which is proven formally by Kaldewaij [Kaldewaij1]. Therefore, we label the states of T with subsets of $\{k \mid 0 \leq k < N\}$. The terminology introduced in this section is used throughout the remainder of this paper.

2 The upper limit

The number of subsets of $\{k \mid 0 \leqslant k < N\}$ equals 2^N. As a consequence, T contains at most 2^N states. If only transitions of type $x \xrightarrow{a} x$ are projected away, no nondeterminism is introduced. In this case, the number of states of T is at most N, the number of states of S. Now, assume that at least one transition of type $x \xrightarrow{a} y$ is projected away, where $x \neq y$. Consider a state in T labeled with a set P that contains x. Due to the introduced nondeterminism, viz. internal move a may or may not have happened, P contains y as well. Therefore, subsets of $\{k \mid 0 \leqslant k < N\}$ that contain x but not y do not occur as the label of some state of T. Since there are $\frac{1}{4} * 2^N$ subsets of $\{k \mid 0 \leqslant k < N\}$ that contain x but not y, T contains at most $(3 * 2^{(N-2)})$ states. Furthermore, the empty set does not occur as the label of some state of T neither. In short, T contains at most $(3 * 2^{(N-2)} - 1)$ states. This proof I owe to Robert Huis in 't Veld.

The set discussed above, which consists of $(3 * 2^{(N-2)} - 1)$ elements, is denoted by L. In short, L consists of all subsets l of $\{k \mid 0 \leqslant k < N\}$, such that l is non-empty and $0 \in l$ implies $1 \in l$.

3 The maximum

In section 3.0 we define state graphs S and T. We show in section 3.1 that all $(3 * 2^{(N-2)} - 1)$ states defined in section 2 be distinct. Finally, section 3.2 deals with the reachability of these states from the initial state.

Notice that we do not need mathematical induction to N, the number of states.

3.0 Definitions

We consider $(N+2)$ distinct symbols : di, for $0 \leqslant i < N$, e, and g. The alphabet A denotes the set $\{g\} \cup \{di \mid 0 \leqslant i < N\}$. State graph S contains the following transitions :

$0 \xrightarrow{e} 1$

$0 \xrightarrow{g} 0$

$k \xrightarrow{g} (k+1)$ for $1 \leqslant k < (N-1)$, and

$k \xrightarrow{di} k$ for $(0 \leqslant k < N) \wedge (0 \leqslant i < N) \wedge (i \neq k)$.

The state labeled 0 is the initial state of S. State graph T is the projection of S onto A. Therefore, the label of the initial state of T is $\{0, 1\}$. In the remainder of this section we consider state graph T. Furthermore, by state P, for P an element of L, we mean the state labeled with P.

Notice that for $i \neq 1$ transition di is possible from each state, say P, such that P contains at least one element besides perhaps i ; $d1$ is possible from each state, say P, such that P contains at least one element besides perhaps 1 and P does not contain 0. From such a state di, $0 \leqslant i < N$, leads to the state $P \setminus \{i\}$, i.e. di "deletes" the element i from the set that labels the state.

3.1 Distinctness

We consider two distinct subsets, say P and Q, of L. Without loss in generality we assume that x be such that $x \in P \setminus Q$. Since Q is nonempty, cf. section 3.0, we choose y in Q, such that $y=1$ if $0 \in Q$. This is possible since $0 \in Q$ implies $1 \in Q$, cf. section 3.0. Let s be a (perhaps empty) sequence of the transitions di, for which $i \in Q \setminus \{y\}$, i.e. s is a permutation of the elements of $Q \setminus \{y\}$. This sequence s leads from state Q to state $\{y\}$; moreover, s leads from state P to some state, say R. Since $x \in P \setminus Q$, $x \neq y$ and $x \in R$ hold. Transition dy is possible from state R as it is not from state $\{y\}$. As a consequence, the sequence $(s ; dy)$, i.e. s followed by dy, of transitions is possible from state P, as it is not from state Q. We conclude that states P and Q be distinct.

3.2 Reachability

Due to the nondeterminism that is introduced by the projection, the path $\{0,1\} \xrightarrow{g^{(N-2)}} \{k \mid 0 \leq k < N\}$, i.e. the path from $\{0,1\}$ via a sequence of $(N-2)$ transitions g to $\{k \mid 0 \leq k < N\}$, exists. Furthermore, it is obvious that all $(3 * 2^{(N-2)} - 1)$ states mentioned in section 3.1 are reachable from state $\{k \mid 0 \leq k < N\}$ by "deleting" symbols from the latter, cf. section 3.0. Combining this with our first observation, we conclude that all $(3 * 2^{(N-2)} - 1)$ states mentioned in section 3.1 are reachable from the initial state $\{0,1\}$.

4 Remarks

Projecting away two transitions, say $h \xrightarrow{a} j$ and $k \xrightarrow{b} l$, yields a state graph with less than $(3 * 2^{(N-2)} - 1)$ states, provided that $h \neq j$, $k \neq l$, and $((h \neq k) \vee (j \neq l))$. This is due to the introduction of more nondeterminism. Since the proof hereof is easy and analogous to the proof of our upper limit, we do not present it here.

In a preliminary version of this paper we suggested to use an alphabet consisting of $(3 * 2^{(N-2)} + 1)$ symbols. By using such large an alphabet it is possible to go directly via a transition from state $\{k \mid 0 \leq k < N\}$ to any particular state. Of course, there exists a trade-off between the number of symbols of the alphabet and the (maximum) length of the path of transitions, that is needed to go from $\{k \mid 0 \leq k < N\}$ to an arbitrary state. The solution presented in section 3 arises from suggestions by Tom Verhoeff.

5 Acknowledgements

Acknowledgements are due to the members of the Eindhoven VLSI Club for discussing the problem and the solution presented above. In particular Robert Huis in 't Veld showed (the existence of) the upper limit, and Tom Verhoeff suggested to use an alphabet consisting of only $(N+2)$ symbols.

References

[Hoare] C.A.R. Hoare. *Communicating Sequential Processes*, Prentice/Hall International, UK, LTD., London, 1985.

[Kaldewaij0] A. Kaldewaij, *A Formalism for Concurrent Processes*, Dissertation, Eindhoven University of Technology, 1986.

[Kaldewaij1] A. Kaldewaij, personal communication.

Eindhoven, March 19, 1987

COMPUTING SCIENCE NOTES

In this series appeared :

No.	Author(s)	Title
85/01	R.H. Mak	The formal specification and derivation of CMOS-circuits
85/02	W.M.C.J. van Overveld	On arithmetic operations with M-out-of-N-codes
85/03	W.J.M. Lemmens	Use of a computer for evaluation of flow films
85/04	T. Verhoeff H.M.J.L. Schols	Delay insensitive directed trace structures satisfy the foam rubber wrapper postulate
86/01	R. Koymans	Specifying message passing and real-time systems
86/02	G.A. Bussing K.M. van Hee M. Voorhoeve	ELISA, A language for formal specifications of information systems
86/03	Rob Hoogerwoord	Some reflections on the implementation of trace structures
86/04	G.J. Houben J. Paredaens K.M. van Hee	The partition of an information system in several parallel systems
86/05	Jan L.G. Dietz Kees M. van Hee	A framework for the conceptual modeling of discrete dynamic systems
86/06	Tom Verhoeff	Nondeterminism and divergence created by concealment in CSP
86/07	R. Gerth L. Shira	On proving communication closedness of distributed layers

86/08	R. Koymans R.K. Shyamasundar W.P. de Roever R. Gerth S. Arun Kumar	Compositional semantics for real-time distributed computing (Inf.&Control 1987)
86/09	C. Huizing R. Gerth W.P. de Roever	Full abstraction of a real-time denotational semantics for an OCCAM-like language
86/10	J. Hooman	A compositional proof theory for real-time distributed message passing
86/11	W.P. de Roever	Questions to Robin Milner - A responder's commentary (IFIP86)
86/12	A. Boucher R. Gerth	A timed failures model for extended communicating processes
86/13	R. Gerth W.P. de Roever	Proving monitors revisited: a first step towards verifying object oriented systems (Fund. Informatica IX-4)
86/14	R. Koymans	Specifying passing systems requires extending temporal logic
87/01	R. Gerth	On the existence of sound and complete axiomatizations of the monitor concept
87/02	Simon J. Klaver Chris F.M. Verberne	Federatieve Databases
87/03	G.J. Houben J.Paredaens	A formal approach to distributed information systems
87/04	T.Verhoeff	Delay-insensitive codes - An overview

87/05	R.Kuiper	Enforcing non-determinism via linear time temporal logic specification.
87/06	R.Koymans	Temporele logica specificatie van message passing en real-time systemen (in Dutch).
87/07	R.Koymans	Specifying message passing and real-time systems with real-time temporal logic.
87/08	H.M.J.L. Schols	The maximum number of states after projection.