ing. We describe an algori
lates Finite State Machine m
into AND-OR graphs. State
graphs does not suffer from
exhaustive search is an NP
demonstrate that random sea
a vaible alternative to mode
debugging and fast analysis
support our conclusions thro
Dekker's two process mutua
Space Shuttle's liquid hydrog

1 Introduction

Formal modelling, analys
tive research areas in softw
doubt that the application o
the software development lif
ability (e.g., the long list of
doubts exist concerning the
methods:

- The cost of writing the
  ferred to below as the *w*
  mathematical expertise
  models.

Figure 1. The s

Figure 1 illustrates the s
marked *saturation* represent
model in which everything
quickly and then a *saturati*
saturation, a level plateau in
search can not discover any n

Assessment methods lack
the following property: the m
ment, the more unique results
no saturation in Figure 1). C
methods that do exhibit a sa
stopping rules, which can be
mal analysis (the running co
formal model when it is very
uncover new results, i.e., afte
countered.

When we use early-stopp
tives—we may conclude tha
further assessment would l
Hence we endorse early stopp
ods that exhibit the following

- *Adequacy*—an adequate
fail to recognize faults i
plored prior to early stop

symbols (included in Σ):

1. A transition in one mac fact that another machin effect of a transition ma other machine.
2. A transition may be trig from another machine, c be to send a message.

The key difference betwe their use as transition inputs message *consumes* the messa to trigger another. But states they trigger; they are good fo sitions.

2.2 NAYO Graph Tran

Figure 3 shows an AND communicating FSM model this type of AND-OR graph following features:

- A set N of undirected N ible nodes.
- A set A of AND-nodes of its YES-edge parents
- A set Y of directed YES

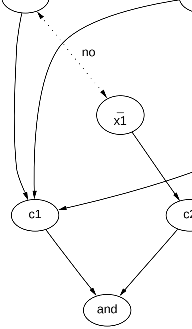

Figure 5. NAYO graph
query $(x_1 \vee x_2 \vee x_3) \wedge ($

Unfortunately, the proble
particular node in the NAYO
complete,[1] which we show h

3SAT $\leq_P$ NAYO search (N
as the 3SAT problem, which
for the 3SAT problem we hav
the conjunction of a series o
disjunction of 3 literals. A lit

[1] *NP* is the class of problems fo
polynomial time (the time required
size); an *NP-complete* problem is (1
class NP and is (2) itself in NP.

Figure 7 shows the rando
plore NAYO graphs. Each ti
its *wait* field is decremente
is *reached*. An OR-node's *w*
once, because we only need
ents; so OR-nodes *wait* field
reach an AND-node, we mu
so its *wait* field is initialized
2).

The central part of the se
4-19. We begin with an inpu
particular order. The first no
5). If it has not been disquali
some node we already belie
explore its children. All chil
ified (line 9). The *wait* fields
are decremented (line 12), a
the way to zero, they are put
dex (line 15). This process
(line 4).

Once all nodes in the *Q* ha
set us up for the next iteratio
nodes marked *true* at the cur
the nodes marked *reached* at
are reached but disqualified,
The *true* set corresponds to t
ure 6, and the *reached* set co

Promela has been designed
gramming language, but repr
SPIN is capable of automatic
machine version of a Promel
model from Figure 9 in this f

Figure 11 shows the result
on a NAYO graph represen
Dekker's mutual exclusion s
der to show that our random
fault, we have added to our
able called *safe*, which is init
proctype A and proctype B a
4 (the critical section). We h
of the following transition to
directly into its critical sectio
and t:

```
state 3 -> sta
```

The searches shown in Fig
of experiments. In every fau
we quickly find all but one o
graph (23 of 24)—we never
the model with the fault, we
24), including the node repr
idea of exactly how *quickly* t
model, the size composite fin
the model that would be sear
this model is bounded at 2,30
our NAYO search reached sa

Dekker model with an err
at what point in a particu
reached.

Each plot shows ten trials o
values; for each trial the sea
many times, each time with
ing track of the total OR-n
the unique OR-nodes reach

**Figure 11. Search
Dekker's solution to t
exclusion problem.**

for—a quick rise to saturatio
remains level indefinitely. W
number of unique OR-node
height 52 (out of 62 total O
and stayed there, and this ha

Why 52? Why were we
OR-nodes? A close look at t
shows three monitored varia
in the environment external
ing on 2 possible values, th
well. For each of these 10 va
NAYO graph—an OR-node
by our search except as part

puter Protocols. Prent
http://cm.bell-lab
spin/Doc/Book91.ht

[6] G. Holzmann. The Mode
tions on Software Enginee

[7] J. Horgan and A. Mathur
ity. In M. Lyu, editor, *The
Engineering*, pages 531–5

[8] H. Kautz and B. Selman
ning, Propositional Log
*Proceedings of the 13th
ficial Intelligence and t
of Artificial Intelligence*
Menlo Park, Aug. 4–8
Press. Available at htt
~jimmyd/summaries/

[9] A. Mackworth and E. Fr
Polynomial Network Cons
Satisfaction Problems.
1985.

[10] T. Menzies and B. Cukic.
ware, 17(5):107–112, 200
menzies.com/pdf/00

[11] T. Menzies, B. Cukic, H. S
determinate Systems. In
http://tim.menzies

[12] T. Menzies and Y. Hu. Ag
editor, *Formal Approache
chapter*, 2002. Availabl
com/pdf/01agents.p