NAME

scrutator – pruning of Labelled Transition Systems

SYNOPSIS

bcg_open [*bcg_opt*] *spec*[**.bcg**] [*cc_opt*] **scrutator** [*scrutator_opt*] *lts*[**.bcg**]

or:

exp.open [*exp_opt*] *spec*[**.exp**] [*cc_opt*] **scrutator** [*scrutator_opt*] *lts*[**.bcg**]

or:

fsp.open [*fsp_opt*] *spec*[**.lts**] [*cc_opt*] **scrutator** [*scrutator_opt*] *lts*[**.bcg**]

or:

lnt.open [*lnt_opt*] *spec*[**.lnt**] [*cc_opt*] **scrutator** [*scrutator_opt*] *lts*[**.bcg**]

or:

lotos.open [*lotos_opt*] *spec*[**.lotos**] [*cc_opt*] **scrutator** [*scrutator_opt*] *lts*[**.bcg**]

or:

seq.open [*seq_opt*] *spec*[**.seq**] [*cc_opt*] **scrutator** [*scrutator_opt*] *lts*[**.bcg**]

DESCRIPTION

scrutator takes as input a Labelled Transition System (LTS) represented either as a BCG graph *spec***.bcg**, a composition expression *spec***.exp**, an FSP program *spec***.lts**, an LNT program *spec***.lnt**, a LOTOS program *spec***.lotos**, or a sequence file *spec***.seq**.

scrutator performs an on-the-fly exploration of the LTS *spec* and prunes certain parts of it according to the options and arguments specified (see OPTIONS below). The resulting LTS, represented as a BCG graph, is stored in the file *lts***.bcg**.

Additionally, **scrutator** can also reduce the LTS on the fly according to various relations (see OPTIONS below).

Note: The method implemented in the current version of **scrutator** (described in [MPS07,MPS12]) is based on a translation of the pruning problem into the resolution of a Boolean Equation System (BES), which is performed on the fly using the algorithms provided by the **caesar_solve_1**(LOCAL) library of OPEN/CAESAR (see the corresponding manual page and the article [Mat06] for details).

OPTIONS

The options *bcg_opt*, if any, are passed to **bcg_lib**(LOCAL).

The options *exp_opt*, if any, are passed to **exp.open**(LOCAL).

The options *fsp_opt*, if any, are passed to **fsp.open**(LOCAL).

The options *lnt_opt*, if any, are passed to **lnt.open**(LOCAL).

The options *lotos_opt*, if any, are passed to **caesar**(LOCAL) and to **caesar.adt**(LOCAL).

The options *seq_opt*, if any, are passed to **seq.open**(LOCAL).

The options *cc_opt*, if any, are passed to the C compiler.

The following options *scrutator_opt* are currently available:

-nodeadlock

Prune *spec* by keeping only the transitions whose target states do not eventually lead to deadlock states.

Formally, a transition s--L-->s' is kept in *lts***.bcg** iff state s' satisfies the following CTL formula:

```
not (AF deadlock)
```

or the equivalent mu-calculus formula:

```
nu X . <true> X
```

This kind of pruning corresponds to the adaptation technique proposed in [CPS06], the difference being that the pruning is performed on-the-fly. This option is mutually exclusive with the **-potential** and **-inevitable** options. Default option.

−potential [−total | −partial | −gate] *matching_filename*

Prune *spec* by keeping only the transitions whose target states potentially lead to states having an outgoing transition labelled by an action matching the rules defined in *matching_filename*.

The format of *matching_filename* is the same as of hiding files defined in the **caesar_hide_1**(LOCAL) manual page, except that the keyword "match" is used instead of "hide". The **-total**, **-partial**, and **-gate** options specify the "total matching", "partial matching", and "gate matching" semantics, respectively. See the **caesar_hide_1**(LOCAL) manual page for more details about these semantics. Option **-total** is the default.

Formally, a transition s--L-->s' is kept in *lts***.bcg** iff state s' satisfies the following CTL formula:

EF <*action***> true**

or the equivalent mu-calculus formula:

mu X . <*action***> true or <true> X**

where *action* denotes an action (transition label) matching the rules specified in *matching_filename*. This kind of pruning corresponds to the adaptation technique proposed in [MPS07,MPS12]. This option is mutually exclusive with the **-nodeadlock** and **-inevitable** options. Not a default option.

−inevitable [−total | −partial | −gate] *matching_filename*

Prune *spec* by keeping only the transitions whose target states eventually lead to states having an outgoing transition labelled by an action matching the rules defined in *matching_filename*.

The format of *matching_filename* is the same as of hiding files defined in the **caesar_hide_1**(LOCAL) manual page, except that the keyword "match" is used instead of "hide". The **-total**, **-partial**, and **-gate** options specify the "total matching", "partial matching", and "gate matching" semantics, respectively. See the **caesar_hide_1**(LOCAL) manual page for more details about these semantics. Option **-total** is the default.

Formally, a transition s--L-->s' is kept in *lts***.bcg** iff state s' satisfies the following CTL formula:

AF <*action***> true**

or the equivalent mu-calculus formula:

mu X . <*action***> true or (<true> true and [true] X)**

where *action* denotes an action (transition label) matching the rules specified in *matching_filename*. This option is mutually exclusive with the **-nodeadlock** and **-potential** options. Not a default option.

-cache *n*
Use a cache of size *n* for storing the states of *spec* during the pruning. This option allows to trade off memory consumption against execution time. Not a default option.

-stat Display statistical information about the resolution of the BES corresponding to the pruning of *spec*. Not a default option.

-version
Display the current version number of the tool and stop. To be effective, this option should occur as the first argument on the command line. Subsequent options and/or arguments, if any, will be discarded. Not a default option.

The options below specify additional reductions that can be applied on-the-fly during the pruning of the LTS:

-tauconfluence
Reduce the LTS on the fly modulo tau-confluence (a form of partial order reduction that preserves branching equivalence). This option can be used in conjunction with options **-taustar** and **-weaktrace**, and in some cases it may reduce the execution time and the memory consumption significantly. Not a default option.

-taustar
Reduce the LTS on the fly modulo tau∗.a equivalence. This reduction eliminates all internal transitions (labelled by the "i" action) in *lts***.bcg**. Not a default option.

-weaktrace
Reduce the LTS on the fly modulo weak trace equivalence. This reduction eliminates all internal transitions and determinizes *lts***.bcg**. Not a default option.

Note: The reduction options above replace the divergences (cycles of tau-transitions) present in the LTS with deadlock states. This may influence the inevitable reachability of certain states (e.g., deadlock states or states having an outgoing transition labelled by a given action), and therefore may trigger a more drastic pruning of the LTS when used in conjunction with the **-nodeadlock** and **-inevitable** options.

EXIT STATUS

Exit status is 0 if everything is alright, 1 otherwise.

BIBLIOGRAPHY

[CPS06]
C. Canal, P. Poizat, and G. Salaun. Synchronizing Behavioural Mismatch in Software Composition. In Roberto Gorrieri and Heike Wehrheim (Eds.), Proceedings of the 8th IFIP International Conference on Formal Methods for Open Object-Based Distributed Systems FMOODS'2006 (Bologna, Italy), Lecture Notes in Computer Science vol. 4037, pp. 63-77. Springer Verlag, June 2006.

[Mat06] R. Mateescu. CAESAR_SOLVE: A Generic Library for On-the-Fly Resolution of Alternation-Free Boolean Equation Systems. Springer International Journal on Software Tools for Technology Transfer (STTT) 8(1):37-56, 2006. Full version available as INRIA Research Report RR-5948.

[MPS07] R. Mateescu, P. Poizat, and G. Salaun. On-the-Fly Adaptation of Component Compositions based on Process Algebra Encodings. In Alexander Egyed and Bernd Fischer (Eds.), Proceedings of the 22nd IEEE/ACM International Conference on Automated Software Engineering ASE'07 (Atlanta, Georgia, USA), pp. 385-388. ACM Press, Nov. 2007. Full version available as INRIA Research Report RR-6362.

[MPS12] R. Mateescu, P. Poizat, and G. Salaun. Adaptation of Service Protocols Using Process Algebra and On-the-Fly Reduction Techniques. IEEE Transactions on Software Engineering 38(4):755-777, 2012.

AUTHORS

Radu Mateescu (INRIA/CONVECS).

OPERANDS

spec.**bcg**	BCG graph (input)
spec.**exp**	network of communicating LTSs (input)
spec.**lts**	FSP specification (input)
spec.**lnt**	LNT specification (input)
spec.**lotos**	LOTOS specification (input)
spec.**seq**	sequence file (input)
lts.**bcg**	BCG graph (output)

FILES

The binary code of **scrutator** is available in $CADP/bin.‘arch‘/scrutator.a

SEE ALSO

bcg(LOCAL), **bcg_open**(LOCAL), **exp**(LOCAL), **exp.open**(LOCAL), **fsp.open**(LOCAL), **lnt.open**(LOCAL), **lotos**(LOCAL), **lotos.open**(LOCAL), **seq**(LOCAL) **seq.open**(LOCAL)

Additional information is available from the CADP Web page located at http://cadp.inria.fr

Directives for installation are given in files **$CADP/INSTALLATION_***.

Recent changes and improvements to this software are reported and commented in file **$CADP/HISTORY**.

BUGS

Please report bugs to Radu.Mateescu@inria.fr