

NAME

bcg_transient – transient numerical analysis of (extended) continuous-time Markov chains encoded in the BCG format

SYNOPSIS

bcg_transient [*bcg_options*] [-**epsilon** *eps*] [-**sol** *solution_file*] [-**thr** [-**append**] *throughput_file*] [-**mat** *matrix_file*] [-**red** *reduced_matrix_file*] [-**log** *log_file*] *filename* [**.bcg**] [*parameter=value ...*] *time_instant ...*

where *bcg_options* is defined below (see GENERAL OPTIONS).

DESCRIPTION

bcg_transient performs transient analysis on *filename.bcg*, which is an (extended) continuous-time Markov chain encoded in the BCG format.

bcg_transient first transforms *filename.bcg* into a numerical matrix indexed by states. Then, it reduces this matrix by normalizing probabilistic transitions, removing unreachable states and "vanishing" states, keeping "tangible" states only (see section INPUT below for details about the BCG graphs accepted by **bcg_transient** and the definition of tangible and vanishing states). As a result, the reduced matrix obtained is the generator matrix of a continuous-time Markov chain. Finally, **bcg_transient** computes the time-dependent ("transient") probability distribution of the model at several (user-specified) time instants using the uniformization algorithm [Jen53] (and the Fox-Glynn method [FG88] to approximate Poisson probabilities). It can also compute throughputs for the transitions of the system.

GENERAL OPTIONS

The following *bcg_options* are currently supported: **-version**, **-create**, **-update**, **-remove**, **-cc**, and **-tmp**. See the **bcg(LOCAL)** manual page for a description of these options.

PARTICULAR OPTIONS

Taking as input *filename.bcg*, on the form of which various restrictions apply (see section INPUT below), and a list of *time_instants*, **bcg_transient** can produce five kinds of output files, depending on the command-line options specified.

The optional list of "*parameter=value*" arguments at the end of the command-line (where *parameter* is any character string that neither contain blanks nor the "=" character, and where *value* is any character string that does not contain blanks) is only meaningful to option **-thr**. These arguments have no influence on the actual numerical computations, they only serve to add columns in throughput tables (see section OUTPUT-2 below).

Time is a special parameter that can be dealt with using the (optional) argument "**time=here**", whose meaning is detailed in section OUTPUT-2 below. When *parameter* is equal to "**time**", *value* must be equal to the keyword "**here**".

The *time_instants* are strictly positive real numbers, and there must be at least one *time_instant*.

The following options are supported:

-epsilon *eps*

Set the precision of certain floating-point comparisons to *eps*, where *eps* is a real value. When *eps* is out of [0..1[, **bcg_transient** reports an error. Default value for *eps* is 1E-6.

-sol *solution_file*

Write the probability vectors, computed at each of the *time_instants* specified on the command-line, to file *solution_file* (see section OUTPUT-1 below for a description of the file format). If *solution_file* already exists, its contents will be overwritten. If *solution_file* is equal to the special string '-', the probability vector is displayed on the standard output. Not a default option.

-thr [-append] *throughput_file*

Write the transition throughputs, computed at each of the *time_instants* specified on the command-line, to file *throughput_file*. The format of this file is determined by the suffix (i.e., file extension) of *throughput_file* (see section OUTPUT-2 below for a description of the available file formats). If *throughput_file* already exists, its contents will be overwritten, unless the **-append** option is specified, in which case the transition throughputs will be added at the end of *throughput_file* so as to form a proper data table. If the **-thr** option is missing or if *throughput_file* is equal to the special string '-', the transition throughputs are displayed on the standard output. Option **"-thr -"** is the default option when the command line does not contain any of the following options: **-mat**, **-red**, **-sol**, and **-thr**.

-mat *matrix_file*

Write the "raw" matrix (prior to matrix reduction) to file *matrix_file*. The format of this file is determined by the suffix (i.e., file extension) of *matrix_file* (see section OUTPUT-3 below for a description of the available file formats). If *matrix_file* already exists, its contents will be overwritten. If *matrix_file* is equal to the special string '-', the matrix is displayed on the standard output. Not a default option.

-red *reduced_matrix_file*

Write the reduced matrix to file *reduced_matrix_file*. The format of this file is determined by the suffix (i.e., file extension) of *reduced_matrix_file* (see section OUTPUT-3 below for a description of the available file formats). If *reduced_matrix_file* already exists, its contents will be overwritten. If *reduced_matrix_file* is equal to the special string '-', the reduced matrix is written on the standard output. Not a default option.

-log *log_file*

Write various information about data structures and computations to file *log_file*. The format of this file is undocumented but self-understandable, and might change in future releases of *bcg_transient*. If *log_file* already exists, its contents will be overwritten. If *log_file* is equal to the special string '-', information is displayed on the standard output. Not a default option.

The files *solution_file*, *throughput_file*, *matrix_file*, *reduced_matrix_file*, and *log_file* should be pairwise different; otherwise, the result is undefined.

INPUT: THE BCG GRAPH

The input of **bcg_transient** is an extended Markovian model combining features from discrete-time and continuous-time Markov chains. In order to be accepted by **bcg_transient**, *filename.bcg* must satisfy several conditions, otherwise an error message will occur

All transition labels of *filename.bcg* must have one of the following forms:

- **"rate %f"** (called a stochastic transition),
- **"label; rate %f"** (called a labelled stochastic transition),
- **"prob %p"** (called a probabilistic transition), or

- "*label*; **prob** %*p*" (called a labelled probabilistic transition) ,

where %*f* denotes a strictly positive floating-point number, %*p* denotes a floating-point number in the range $[0..1]$, and *label* denotes a character string that does not contain the ";" character (*label* may be equal to the internal action, often noted "i" or "tau").

Note: transitions labelled with only "*label*" are always forbidden by **bcg_transient**, including the case where "*label*" denotes the internal action.

When constructing the "raw" matrix, all *labels* occurring in labelled probabilistic transitions are discarded.

If there exists a (labelled) probabilistic transition from a state *S1* to a state *S2*, then all (labelled) stochastic transitions from *S1* to any state (including *S2*) are discarded when constructing the "raw" matrix. This reflects that probabilistic transitions are instantaneous, while stochastic transitions are not.

We classify states as being either *vanishing* if at least one (possibly labelled) probabilistic transition goes out of these states, or *tangible* otherwise.

The input BCG graph should contain at least one tangible state, and it should not contain cycles of vanishing states connected by (possibly labelled) probabilistic transitions.

Note: The sum of %*p* values on all (possibly labelled) probabilistic transitions leaving a vanishing state needs not be equal to 1; if this sum is different from 1, then probabilistic values will be normalized (i.e., divided by this sum).

To build the reduced matrix, **bcg_steady** eliminates all vanishing states, so that this matrix contains tangible states only.

See also **bcg_min**(LOCAL) for a discussion about the various probabilistic and stochastic models present in the scientific literature.

OUTPUT-1: THE SOLUTION VECTOR

The format of the file generated using the **-sol** option of **bcg_transient** is the following. There is one line per time instant and per tangible state. Each line contains three numbers: a real number giving the time instant, an integer corresponding to the state number in the input BCG graph, and a real number corresponding to the probability of being in this state at that time instant.

OUTPUT-2: THE TRANSITION THROUGHPUTS

The throughput table has two (possibly empty) groups of columns:

- The first group contains at least one column for time values, plus one column for each option *parameter=value* given on the command-line (except for the "**time=here**" option, if present, since there is only one single column for time). These columns, if any, are useful when evaluating the performance of a system parameterized with one or more variables, namely to aggregate in the same table the different throughputs measures corresponding to different time instants and different values of the parameters. Columns of the first group appear in the same order as the corresponding options on the command-line. By default, time appears in the first column, unless the "**time=here**" option is present, in which case the time column appears at the place specified by this option.

- The second group contains one column per labelled stochastic transition label present in the input BCG graph, precisely, one column for each different *label* occurring on a transition of the form "*label; rate %f*".

The throughput table starts with a first "header" line followed by one or several "subsequent" lines.

- The header line contains the "titles" of columns. For a column of the first group associated to a *parameter=value* option, the corresponding title is *parameter*, the title of the time column being **time**. For a column of the second group associated to a label, the corresponding title is the label itself.
- Each of the subsequent lines is associated to a particular time instant *t* specified on the command-line and contains values. For a column of the first group associated to a *parameter=value* option: if *parameter* is equal to **time** the corresponding cell contains the value of *t*; otherwise, the corresponding cell contains *value*, which is the same for all time instants. For a column of the second group associated to a *label*, the corresponding cell contains the throughput for this label at time *t*, i.e., the sum, for each stochastic transition "*label; rate %f*", of the rate value *%f* weighted with the probability of being in the tangible source state of this transition at time *t*.

If the **-append** option is absent, or if the throughput file is equal to the special string '-', or if the throughput file does not exist, or if it is empty, *bcg_transient* will generate automatically the header line and the subsequent lines.

Otherwise, the first line of the throughput file is expected to contain the titles of columns and will be parsed to identify the correspondance between labels and columns. In particular, **bcg_transient** checks that the first group of columns corresponds to the parameters given on the command-line. After parsing the header line, **bcg_transient** will append the subsequent lines at the end of the throughput file. As regards the second group of columns, if the label of a given column title does not occur in *filename.bcg*, a zero throughput will be reported in the corresponding column; conversely, labels of *filename.bcg* for which there is no corresponding column title will be ignored.

Throughputs can be displayed in two different formats, which are determined according to the suffix (i.e., file extension) of the throughput file name.

- If the file name has the ".csv" extension, the throughput table will be displayed in the CSV (Comma-Separated Values) exchange format understood by most relational data base applications and spreadsheet software (such as Microsoft Excel, etc.).
- Otherwise, if the file name has a different extension, or no extension, or if it is the standard output, throughputs will be displayed in a human-readable format that is essentially the same format as CSV with commas replaced by spaces so as to align columns properly. Note that this format is also understood by some data visualization tools such as Gnuplot.

OUTPUT-3: THE MATRICES

Both the "raw" matrix produced using option **-mat** and the reduced matrix produced using option **-red** follow the same format conventions. The essential difference is that the former contains vanishing and tangible states, whereas the latter only contains tangible states. Also, the reduced matrix is a generator of a continuous-time Markov chain.

For two different indexes i and j , the element (i,j) of the matrix, located at the i -th row and the j -th column, is the sum of all the floating-point numbers associated to the (labelled) stochastic or probabilistic transitions going from the j -th state to the i -th state, where floating-point numbers associated to (labelled) stochastic transitions are interpreted as positive numbers whereas floating-point numbers associated to (labelled) probabilistic transitions are interpreted as negative numbers between -1 and 0. Note that rates and probabilities are never mixed since, between two states, there cannot be stochastic and probabilistic transitions at the same time.

The diagonal elements (i,i) are defined to be the negative sum of all matrix elements (i,j) with j different from i .

Matrices can be displayed in three different formats, which are determined according to the suffix (i.e., file extension) of the matrix file name.

- If the file name has the ".csv" extension, the matrix will be displayed in the CSV (Comma-Separated Values) format mentioned above. Each row of the matrix is displayed on one line of the output file, and on each line, the matrix elements are separated with commas.
- If the file name has the ".spm" extension, the matrix will be displayed in the format used by the Sparse 1.3 software library (see the CREDITS section below). This format is the following. The first line of the file contains the file name. The second line contains the number of states, followed by the "**real**" keyword. Then, there is one line per each non-zero element (i,j) in the matrix. Each line contains two integers followed by one real number: the value of i , the value of j , and the value of matrix element (i,j) . The file ends with a "sentinel" line consisting of three zeros.
- Otherwise, if the file name has a different extension, or no extension, or if it is the standard output, the matrix will be displayed in a human-readable format. The columns of the matrix are split into "packets" so that the text fits on the size of the display. The indexes of rows and columns are indicated and null elements of the matrix are displayed as "..." instead of "0". Statistics (such as matrix size, density, etc.), are displayed after the matrix.

Note: for graphs with many states, whatever the chosen matrix format, the matrix files can be large and writing them to disk may take time.

ENVIRONMENT VARIABLES

See the **bcg(LOCAL)** manual page for a description of the environment variables used by all the BCG application tools.

EXIT STATUS

Exit status is 0 if everything is all right, 1 otherwise.

AUTHORS

The first version of **bcg_transient** was written by Christophe Joubert (INRIA/VASY) and Holger Hermanns (Saarland University and University of Twente). The algorithm for uniformization is based on a former implementation by Vassilis Mertsiotakis (University of Erlangen). Bruno Oudet (INRIA/VASY) ported the tool to various architectures. David Champelovier and Hubert Garavel (both at INRIA/VASY) deeply revised the **bcg_transient** code and manual page to allow their integration within CADP. Holger Hermanns and Frederic Lang (INRIA/VASY) proof-checked the manual page.

CREDITS

bcg_transient uses (an extended version of) the Sparse 1.3 package from the University of California, Berkeley, developed by Kenneth S. Kundert under the supervision of Alberto Sangiovanni-Vincentelli.

OPERANDS

<i>filename.bcg</i>	BCG graph (input)
<i>filename@1.o</i>	dynamic library (input or output)

FILES

\$CADP/bin.arch/bcg_transient “**bcg_transient**” program

See the **bcg(LOCAL)** manual page for a description of the other files.

SEE ALSO

bcg(LOCAL), **bcg_min(LOCAL)**, **bcg_steady(LOCAL)**, **determinator(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 cadp@inria.fr

BIBLIOGRAPHY

[FG88] B.L. Fox and P.W. Glynn. Computing Poisson probabilities. Communications of the ACM 31(4), 1998, pages 440-445.

[GH02] H. Garavel and H. Hermanns. On Combining Functional Verification and Performance Evaluation using CADP. In proceedings of FME'2002, LNCS 2391, pages 410-429, Springer Verlag. Full version available as INRIA Research Report 4492. Available from <http://cadp.inria.fr/publications/Garavel-Hermanns-02.html>

[HJ03] H. Hermanns and Ch. Joubert. A Set of Performance and Dependability Analysis Components for CADP. In proceedings of TACAS'2003, LNCS 2619, pages 425-430, Springer Verlag. Available from <http://cadp.inria.fr/publications/Hermanns-Joubert-03.html>

[Jen53] A. Jensen. Markov Chains as an Aid in the Study of Markov Processes. Skand. Aktuarietidskrift 3, pages 87-91, 1953.

[Mer98] V. Mertsotakis. Approximate Analysis Methods for Stochastic Process Algebras. Ph.D Thesis, University of Erlangen (Germany).