Kind 2

A multi-engine, parallel, SMT-based automatic model checker for safety properties of Lustre programs.

Kind 2 takes as input a Lustre file annotated with properties to prove invariant (see Lustre syntax), and outputs which of the properties are true for all inputs, as well as an input sequence for those properties that are falsified. To ease processing by front-end tools, Kind 2 can output its results in XML format.

Kind 2 runs a process for bounded model checking (BMC), a process for k-induction, and a process for IC3 in parallel on all properties simultaneously. It incrementally outputs counterexamples to properties as well as properties proved invariant.

The following command-line options control its operation (run Usage: kind2 [options] FILEProve properties in Lustre program FILE –timeout_wall (default: 0) Wallclock timeout for the analysis of lowest level. - in modular mode, specifies the timeout for the analysis of ONE (sub)system. - in compositional mode, specifies the timeout for the analysis in one abstraction configuration. –timeout_virtual (default: 0) CPU timeout. –smtsolver (available: Z3, CVC4, MathSat5, Yices, Yices2, default: detect) SMT solver used during the analysis. –smtlogic (available: none, detect, logic, default: none) select logic for SMT solvers (none for no logic (default), and detect to detect with the input system, other SMTLIB logics will be passed to the solver). –z3_bin (default: z3) Executable for the z3 solver. –cvc4_bin (default: cvc4) Executable for the CVC4 solver. –mathsat5_bin (default: mathsat) Executable for the MathSAT5 solver. –yices_bin (default: yices) Executable for the Yices solver. –yices2_bin (default: yices-smt2) Executable for the Yices2 solver. –smt_trace Write all SMT commands to files –smt_trace_dir (default: /Volumes/home/uchuu/repos/kind2/doc/usr) Directory for trace logs of SMT commands. –enable (available: PDR, BMC, IND, INVGEN, INVGENOS, interpreter, default: [BMC, IND, PDR, INVGEN, INVGENOS]) Enables a Kind module. –modular (default: false) Activates bottom up modular analysis. –modular_timeout (modular, default: 0) Wallclock timeout for each subsystem of the modular analysis. –compositional (contracts, default: false) Activates abstraction of subnodes during analysis of a node. If the analysis does not succeed, the ‘abstraction depth’ is lowered, i.e. previously abstracted nodes are not abstracted anymore but their subnodes are. Only nodes with a contract can/will be abstracted. –contracts_subreqs (contracts, default true) Activates the verification of subnode requirements. –version Output version information and exit. –bmc_check (BMC, default: true) BMC will check at each k that the system has reachable states. –bmc_max (BMC, IND, default: 0, unlimited: 0) Maximal number of iterations. –ind_compress (IND, default: false) Compress inductive counterexamples. –ind_compress_equal (IND, default: true) Compress inductive counterexamples for states equal modulo inputs. –ind_compress_same_succ (IND, default: false) Compress inductive counterexamples for states with same successors. –ind_compress_same_pred (IND, default: false) Compress inductive counterexamples for states with same predecessors. –ind_print_inductive_cex (IND, default: true) Print inductive counterexamples. –pdr_qe (PDR, available: Z3, Z3-impl, Z3-impl-2, cooper, default: cooper) Choose quantifier elimination algorithm. –pdr_extract (PDR, available: first, vars, default: first) Heuristics for extraction of implicant. –pdr_check_inductive (PDR, default: true) Check inductiveness of blocking clauses. –pdr_fwd_prop_check_multi (PDR, default: false) Simultaneous check for forward propagation. –pdr_print_inductive_assertions (PDR, default: false) Output inductive blocking clauses. –pdr_print_blocking_clauses (PDR, default: false) Output all blocking clauses. –pdr_print_to_file (PDR, default: stdout) Output file for blocking clauses. –pdr_tighten_to_unsat_core (PDR, default: true) Tighten blocking clauses to an unsatisfiable core. –pdr_inductively_generalize (PDR, default: 1) Inductively generalize blocking clauses before forward propagation (0 = none, 1 = normal IG, 2 = IG with ordering). –pdr_block_in_future (PDR, default: true) Block counterexample in future frames. –pdr_print_inductive_invariant (PDR, default: true) Print inductive invariant if property proved. –pdr_check_inductive_invariant (PDR, default: true) Check inductive invariant if property proved. –cooper_order_var_by_elim (Cooper QE, available: true, false, default: false) Order variables in polynomials by order of elimination. –cooper_general_lbound (Cooper QE, available: true, false, default: false) Choose lower bounds containing variables. –testgen_len (default: 5) Maxmimum length for test generation. –invgen_prune_trivial (default: true) Invariant generation will only look for top node invariants. –invgen_max_succ (default: 1) Maximal number of successive iterations for subsystems. –invgen_lift_candidates (default: false) Invariant generation will only look for top node invariants. –invgen_top_only (default: false) Invariant generation will only look for top node invariants. –invgen_mine_trans (default: false) Invariant generation will extract candidate terms from the transition predicate. –invgen_renice (default: 0) Renice invariant generation process. Give a positive argument to lower priority. –interpreter_input_file (Interpreter) The interpreter will read inputs from this file. –interpreter_steps (Interpreter, default: 0) Run number of steps, override the number of steps given in the input file. –lustre-main (default: –%MAIN annotation) Use the given node as top node in Lustre input. –input-format (available: lustre, native, default: lustre) Format of input file. –debug Enable debug output for a section, give one –debug option for each section to be enabled –debug-log (default: stdout) Output debug messages to file. -s Silence output, errors only. -q Disable output, fatal errors only. -qq Disable output completely. -v Output informational messages. -vv Output informational and debug messages. -vvv Output informational, debug and trace messages. -xml Output in XML format. -help Display this list of options –help Display this list of options -h Display this list of options for a full list).

Select model checking engines

By default, all three model checking engines are run in parallel. Give any combination of , and to select which engines to run. The option –enable BMC alone will not be able to prove properties valid, choosing –enable IND only will not produce any results. Any other combination is sound (properties claimed to be invariant are indeed invariant) and counterexample-complete (a counterexample will be produced for each property that is not invariant, given enough time and resources).

Run for SECS seconds of wall clock time

Run for SECS of CPU time

Select SMT solver

The default is , but see options of the script to override at compile time

Executable for Z3

Executable for CVC4

Executable for MathSat5

Run bounded model checking for up to steps

Output informational messages

Output in XML format

Requirements

- Linux or Mac OS X,
- OCaml 4.02 or later,
- Camlp4
- Menhir parser generator, and
- a supported SMT solver
- Z3 (presently recommended),
- CVC4, (must use) or
- MathSat5

Building and installing

If you got the sources from the Github repository, you need to run first

```
./autogen.sh
```

By default, kind2: No input file given will be installed into , an operation for which you usually need to be root. Call

```
./build.sh --prefix=PATH
```

to install the Kind 2 binary into . You can omit the option to accept the default path of .

The ZeroMQ and CZMQ libraries, and OCaml bindings to CZMQ are distributed with Kind 2. The build script will compile and link to those, ignoring any versions that are installed on your system.

If it has been successful, call

```
make install
```

to install the Kind 2 binary into the chosen location. If you need to pass options to the configure scripts of any of ZeroMQ, CZMQ, the OCaml bindings or Kind 2, add these to the call. Use after to see all available options.

You need a supported SMT solver, at the momemt either Z3, CVC4 or MathSat5 on your path when running kind2: No input file given.

Apache License Version 2.0, January 2004 http://www.apache.org/licenses/

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

1. Definitions.

   License shall mean the terms and conditions for use, reproduction, and distribution as defined by Sections 1 through 9 of this document.

   Licensor shall mean the copyright owner or entity authorized by the copyright owner that is granting the License.

   Legal Entity shall mean the union of the acting entity and all other entities that control, are controlled by, or are under common control with that entity. For the purposes of this definition, control means (i) the power, direct or indirect, to cause the direction or management of such entity, whether by contract or otherwise, or (ii) ownership of fifty percent (50%) or more of the outstanding shares, or (iii) beneficial ownership of such entity.

   You (or Your) shall mean an individual or Legal Entity exercising permissions granted by this License.

   Source form shall mean the preferred form for making modifications, including but not limited to software source code, documentation source, and configuration files.

   Object form shall mean any form resulting from mechanical transformation or translation of a Source form, including but not limited to compiled object code, generated documentation, and conversions to other media types.

Work shall mean the work of authorship, whether in Source or Object form, made available under the License, as indicated by a copyright notice that is included in or attached to the work (an example is provided in the Appendix below).

Derivative Works shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

Contribution shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, submitted means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as Not a Contribution.

Contributor shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent

infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

   (a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

   (b) You must cause any modified files to carry prominent notices stating that You changed the files; and

   (c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

   (d) If the Work includes a NOTICE text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

   You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as

required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing the Work or Derivative Works thereof, You may choose to offer, and charge a fee for, acceptance of support, warranty, indemnity, or other liability obligations and/or rights consistent with this License. However, in accepting such obligations, You may act only on Your own behalf and on Your sole responsibility, not on behalf of any other Contributor, and only if You agree to indemnify, defend, and hold each Contributor harmless for any liability incurred by, or claims asserted against, such Contributor by reason of your accepting any such warranty or additional liability.

END OF TERMS AND CONDITIONS

APPENDIX: How to apply the Apache License to your work.

```
To apply the Apache License to your work, attach the following
boilerplate notice, with the fields enclosed by brackets {}
replaced with your own identifying information. (Don't include
the brackets!)  The text should be enclosed in the appropriate
comment syntax for the file format. We also recommend that a
file or class name and description of purpose be included on the
same printed page as the copyright notice for easier
identification within third-party archives.
```

Copyright {yyyy} {name of copyright owner}

Licensed under the Apache License, Version 2.0 (the License); you may not use this file except in compliance with the License. You may obtain a copy of the License at

```
http://www.apache.org/licenses/LICENSE-2.0
```

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an AS IS BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.