MINTS (MINimizer for Test Suites)

Prerequisites:

For MINTS to work properly, *timeout* and constraint solvers intended to be used must be installed separately. In other words, these executables must be accessible through the $PATH environment variable. In this release of MINTS, following solvers are supported: MiniSat+ v1.0, bsolo v3.0.15, PBS v4, OPBDP v1.1, glpPB v0.2, and Pueblo 1.5.

Installation:

1) Extract contents of the MINTS package tgz file.
2) Set up an environment variable $MINTS, and point $MINTS to the folder extracted in the previous step.

Usage:

Follow instructions in the file format section to set up input files and use corresponding script file to invoke MINTS.

Given a minimization policy defined in PRIORITY, execute the following command:

$MINTS/mints_(solver executable name).sh (solver executable) (time-out limit in seconds) < PRIORITY

Testing MINTS:

A set of example inputs can be founded in $MINTS/tests.

For example, to test MINTS with the bsolo constraint solver, one can execute the following command in the $MINTS/tests folder:

$MINTS/mints_bsolo.sh bsolo 10 < test_priority

The previous command invokes *bsolo* as the back-end solver with a 10-second time-out limit for each of the prioritized relative criteria.

Check text files "solver_output_*i*", *i* starts from 1 to the number of prioritized relative criteria. Each solver_output_*i* file is generated by the back-end solver (in this case, *bsolo*) for the corresponding prioritized criteria. The last solver_output_*i* file consists of an assignment to variables. Such a assignment provides an example of the optimized test suite regarding to the policy specified in test_priority. Please refer to the manual of each individual solver for details of output file formats.

MINTS Inputs Format

A complete policy description consists of five types of files, which include *one* PRIORITY file, *one* or *several* BATCH files, *one* or *several* RELATIVE_CRITERIA files, *one* or *several* ABSOLUTE_CRITERIA files, and *one* or *several* RELATIVE_CRITERION files.

This section describes how to specify a test-suite optimization policy with each file type in following paragraphs.

PRIORITY

PRIORITY is a ascii text file of one or multiple lines. In PRIORITY, each line stores the absolute path to a BATCH file. Each BATCH file describes a weighted objective group of a hybrid policy depicted in section 4.3 of the MINTS paper, and the line number indicates order of the corresponding objective group.

If a tester is interested in specifying a weighted policy, then she needs only one BATCH file. On the other hand, if she is interested in specifying a prioritized policy or a hybrid policy, then she needs to specify *k* BATCH files, where *k* is the number of objectives in a prioritized policy, or the number of objective groups in a hybrid policy.

BATCH

A BATCH file stores information regarding a weighted objective group in a hybrid optimization policy. BATCH is a ascii text file consisting of four lines. The first line stores a natural number, the number of test cases in the test suite to be optimized. The second line is the absolute path to a RELATIVE_CRITERIA file. The third line stores the number of absolute minimization criteria to be considered, as depicted in section 4.4 of the MINTS paper. The last line stores the absolute path to the ABSOLUTE_CRITERIA file, the absolute criteria considered for current weighted objective group.

RELATIVE_CRITERIA

A RELATIVE_CRITERIA file keeps track of relative criteria considered. RELATIVE_CRITERIA is a ascii text file of multiple lines, of which each line stores the absolute path to a RELATIVE_CRITERION file.

RELATIVE_CRITERION

A RELATIVE_CRITERION file stores the weight assigned to a relative criterion. RELATIVE_CRITERION is a ascii text file of two lines, of which the first line stores a integer, the weight assigned to the relative criterion, and the second line stores coefficients of the relative criterion. Given that the number of test cases in the BATCH file is stated as t, the second line in RELATIVE_CRITERION consists of t integers separated with the white space character '\ '.
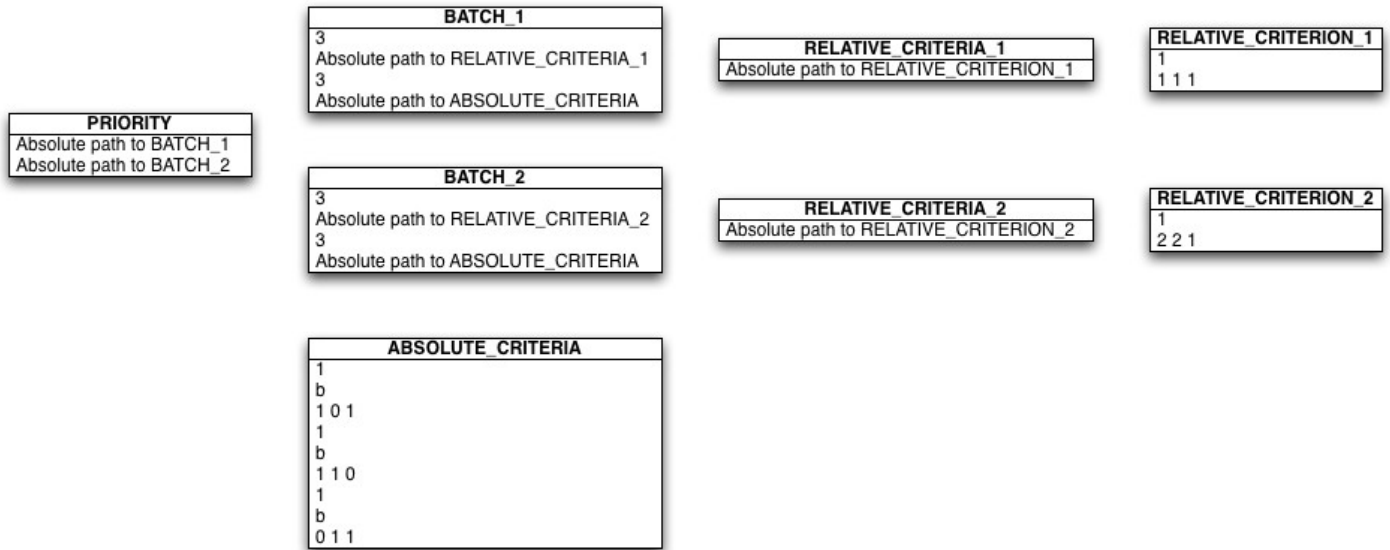

ABSOLUTE_CRITERIA

ABSOLUTE_CRITERIA is a ascii text file of multiple lines. A ABSOLUTE_CRITERIA file stores the set of absolute criteria that must be satisfied. Each absolute criterion occupies three line of space. The first line consists of a single character, ranging from 'a' to 'e'. This character indicates the type of binary operator in the absolute criterion: 'a' stands for '>', 'b' stands for '>=', 'c' stands for '=', 'd' stands for '<=', and 'e' stands for '<'. The second line stores an integer value, the right hand side of the absolute criterion. The third line stores the coefficients of the left hand side of the absolute criterion, in the same format of the second line of a RELATIVE_CRITERION file.

Example 1

Consider a test suite of three test cases. The test subject consists of three lines of code. The first test case covers the first two lines, the second test covers the last two lines, and the third test covers the first and the last line. The first two test cases require two seconds to complete, and the third test only requires one second.

Suppose we would like to specify a prioritized policy, in which the first relative criterion is to minimize the size of the test suite, and the second relative criterion is to minimize the execution time of the test suite. The absolute criteria applied to all relative criteria are that every line of code should be executed at least once in the optimized test suite.

The MINTS input of this test suite for the weighted policy can be expressed in eight files: PRIORITY, BATCH_1, BATCH_2, RELATIVE_CRITERIA_1, RELATIVE_CRITERIA_2, RELATIVE_CRITERION_1, RELATIVE_CRITERION_2, and ABSOLUTE_CRITERIA. Contents of each file is shown in the following graph.

BATCH_1
```
3
Absolute path to RELATIVE_CRITERIA_1
3
Absolute path to ABSOLUTE_CRITERIA
```

RELATIVE_CRITERIA_1
```
Absolute path to RELATIVE_CRITERION_1
```

RELATIVE_CRITERION_1
```
1
1 1 1
```

PRIORITY
```
Absolute path to BATCH_1
Absolute path to BATCH_2
```

BATCH_2
```
3
Absolute path to RELATIVE_CRITERIA_2
3
Absolute path to ABSOLUTE_CRITERIA
```

RELATIVE_CRITERIA_2
```
Absolute path to RELATIVE_CRITERION_2
```

RELATIVE_CRITERION_2
```
1
2 2 1
```

ABSOLUTE_CRITERIA
```
1
b
1 0 1
1
b
1 1 0
1
b
0 1 1
```

Example 2

Consider the same test suite in the previous example. Suppose we would like to specify a weighted policy, in which the first relative criterion is to minimize the size of the test suite, and the second relative criterion is to minimize the execution time of the test suite. The two relative criteria are assigned with equal weight. The absolute criteria applied to all relative criteria are that every line of code should be executed at least once in the optimized test suite.

The input to MINTS of this test suite for the weighted policy is specified using six files: PRIORITY, BATCH, RELATIVE_CRITERIA, RELATIVE_CRITERION_1, RELATIVE_CRITERION_2, and ABSOLUTE_CRITERIA. Contents of each file is shown in the following graph.

PRIORITY
```
Absolute path to BATCH
```

BATCH
```
3
Absolute path to RELATIVE_CRITERIA
3
Absolute path to ABSOLUTE_CRITERIA
```

RELATIVE_CRITERIA
```
Absolute path to RELATIVE_CRITERION_1
Absolute path to RELATIVE_CRITERION_2
```

RELATIVE_CRITERION_1
```
1
1 1 1
```

RELATIVE_CRITERION_2
```
1
2 2 1
```

ABSOLUTE_CRITERIA
```
1
b
1 0 1
1
b
1 1 0
1
b
0 1 1
```