

The folder 2_executionTestCases contains three csv files with the results of the experiments, one per each debugging techniques:

- MRDebug-FL: contains the aggregate results of the fault localization technique.
- MRDebug-IR: contains the aggregate results of the input reduction technique.
- MRDebug-IR-FL: contains the aggregate results of the combination of both techniques.

This README file describe the columns of MRDebug-FL.csv, MRDebug-IR.csv and MRDebug-IR-FL.csv. Each one is detailed in the below subsections.

MRDebug-FL.csv file

MRDebug-FL.csv contains one row per each experimentation unit. The number of columns are 5170, and they are the following:

- Program.name: name of the program executed
- Blocking.factor.input.data: the blocking factor of the test case executed.
- Number.of.test.case: number of the test case executed. The test case can be found in 1_testCases.zip under the folder Program.name/Blocking.factor.input.data/ with the Number.of.test.case identifier
- Number.of.scenarios.to.be.generated.per.characteristic: the number of K executed in the fault localization
- Number.of.input.data.before.localization: the number of <key, value> pairs of the test case
- Seed: the seed used in the fault localization to control the randomness and guarantee the reproduction of the experiments
- Has.Combiner: boolean that indicates if the program executes Combiner function or not
- Execution.time:
 - Execution.time.of.testing.phase.ms: the number of ms employed during the testing phase just before start the fault localization
 - Execution.time.of.localization.ms: the number of ms employed during the fault localization phase
- Scenarios:
 - Scenarios.generated.during.localization: number of configurations executed during the fault localization
 - Scenarios.that.produce.failure.generated.during.localization: number of configurations that triggers a failure
 - Scenarios.that.produce.succeed.generated.during.localization: number of configurations that masks the fault
- spectra: per each characteristic CH:
 - spectra_CH_ncf: number of configurations covered during the fault localization that triggers failure
 - spectra_CH_nuf: number of configurations uncovered during the fault localization that triggers failure
 - spectra_CH_ncs: number of configurations covered during the fault localization that masks the fault
 - spectra_CH_nus: number of configurations uncovered during the fault localization that masks the fault
 - spectra_CH_nc: number of configurations covered during the fault localization

- spectra_CH_nu: number of configurations uncovered during the fault localization
 - spectra_CH_ns: number of configurations that masks the fault
 - spectra_CH_nf: number of configurations that triggers failure
- Ranking positions: per each ranking metric RM and characteristic CH:
 - RM_CH_suspiciousness: the suspiciousness of the characteristic CH given by the ranking metric RM in the fault localization
 - RM_CH_position: the position of CH in the suspiciousness ranking according to the fault localization:
 - RM_CH_position_best: using the tie-breaking best
 - RM_CH_position_worst: using the tie-breaking worst
 - RM_CH_position_random: using the tie-breaking randomly
 - RM_CH_position_average: using the tie-breaking average
- Ranking of the root cause of the fault: position and score of the root cause of the fault per each ranking metric RM
 - RM_position_root_cause: the position of the root cause of fault in the ranking
 - RM_position_root_cause_best: using the tie-breaking best
 - RM_position_root_cause_worst: using the tie-breaking worst
 - RM_position_root_cause_random: using the tie-breaking randomly
 - RM_position_root_cause_average: using the tie-breaking average
 - RM_exam_score: exam score of the root cause of the fault using the ranking metric RM in the fault localization
 - RM_exam_score_best: using the tie-breaking best
 - RM_exam_score_worst: using the tie-breaking worst
 - RM_exam_score_random: using the tie-breaking randomly
 - RM_exam_score_average: using the tie-breaking average

MRDebug-IR.csv file

MRDebug-FL.csv contains one row per each experimentation unit in either MRDebug-IR or the baseline. The number of columns are 13, and they are the following:

- Program.name: name of the program executed
- Blocking.factor.input.data: the blocking factor of the test case executed.
- Number.of.test.case: number of the test case executed. The test case can be found in 1_testCases.zip under the folder Program.name/Blocking.factor.input.data/ with the Number.of.test.case identifier
- Input.Reduction.technique: indicates if the input reduction is executed or if it a random reduction
- Seed: the seed used in the fault localization to control the randomness and guarantee the reproduction of the experiments
- Has.Combiner: boolean that indicates if the program executes Combiner function or not
- Number.of.input.data:
 - Number.of.input.data.before.input.reduction: number of <key, value> pairs of the test case
 - Number.of.input.data.after.input.reduction: number of <key, value> pairs of the test case after executing the test input reduction
- Execution.time:

- Execution.time.of.testing.phase.ms: the number of ms employed during the testing phase just before start the fault localization
- Execution.time.of.input.reduction.phase.ms: the number of ms employed during the input reduction phase
- Execution.time.of.testing.and.input.reduction.ms: the number of ms employed during both testing and input reduction
- Seed.of.scenario.detects.that.detects.fault.in.reduction: the input reduction obtains a minimal configuration that triggers the failure in some executions. This seed guarantees the reproduction of the failure
- Number.of.individuals.generated: number of subsets of test input data analysed during the input reduction

MRdebug-IR-FL.csv file

MRDebug-IR-FL.csv contains one row per each experimentation unit in each subset of <key, value> pairs reduced during the input reduction. The number of columns are 5176 and they are the same of MRDebug-FL and MRDebug-IR, but with the following differences:

- Individuals:
 - Maximum.number.of.individuals.that.can.be.achieved.in.reduction: maximum number of subsets analyzed by the input reduction to achieve the maximum reduction of MRDebug-IR
 - Number.of.individual.that.generates.the.reduction: number of the individual of the current row. For example, if the individual is 2, means that the current row contains the execution of the MRDebug-IR-FL executed after analyze 2 subsets of <key, value> pairs during the input reduction. In contrast, if the number of individual is equal to the maximum, then the current row contains the execution of MRDebug-IR-FL reducing the test input data until the maximum
- Number.of.data:
 - Number.of.data.before.reduction: the number of <key, value> pairs of the test input data
 - Number.of.data.after.reduction: the number of <key, value> pairs after the input reduction and before the fault localization given the current individual