

The ltest Package

**An implementation of Barmak and
Minian's I-test.**

Version 1.0.1

2018

Iván Sadofski Costa

Iván Sadofski Costa Email: isadofski@dm.uba.ar

Homepage: <http://mate.dm.uba.ar/~isadofski>

Copyright

© 2018 Iván Sadofski Costa.

Contents

1	The I-test	4
1.1	Functions for the I-test	4
2	Installing and Loading the Itest Package	7
2.1	Required Packages	7
2.2	Loading the Itest Package	7
	References	8
	Index	9

Chapter 1

The I-test

The I-test is an asphericity test that is not based on a non-positive curvature argument. If a finite group presentation passes the I-test, then the associated 2-complex is diagrammatically reducible (DR), in particular it is aspherical. This test was introduced by Barmak and Minian [BM18, Theorem 2.6]. The main function of the package is `ITest` (1.1.6).

See Sections 2.1 and 2.2 for how to install and load the `ltest` package.

1.1 Functions for the I-test

The following functions are available:

1.1.1 SubwordMatrix

▷ `SubwordMatrix(G)` (function)

G must be an `FpGroup`. Returns the subword matrix of G .

Example

```
gap> F:=FreeGroup(["x","y","z","w"]);;
gap> AssignGeneratorVariables(F);
#I Assigned the global variables [ x, y, z, w ]
gap> G:=F/[x^2*y^2*z^2, x*y*x^-1*z*y*z^-1, w^2*x^-1*w^-1*z];;
gap> SubwordMatrix(G);
[
  [[x^2*y^2*z^2,x*y^2*z^2],[x*y*x^-1*z*y*z^-1,z*y*z^-1],[w^-1*z]
  ],
  [[y^2*z^2,y*z^2],[y*x^-1*z*y*z^-1,y*z^-1],[]
  ],
  [[z^2,z],[z*y*z^-1,<identity ...>],[z]
  ],
  [[],[],[w^2*x^-1*w^-1*z,w*x^-1*w^-1*z,z]]
]
```

1.1.2 WeightMatrix

▷ `WeightMatrix(G, v)` (function)

G must be an `FpGroup`, v a vector orthogonal to $q(r)$ for each relator r of G . Returns the weight matrix of G .

Example

```
gap> M:=WeightMatrix(G,[1,0,-1,2]);
[
  [ 0,-1], [ 0,0], [-3]      ],
  [ -2,-2], [-1,1], []       ],
  [ -2,-1], [ 0,0], [-1]     ],
  [ [],      [],    [0,-2,-1] ]
]
```

1.1.3 IsGoodMatrix

▷ IsGoodMatrix(M)

(function)

Returns true if the matrix M is *good* in the sense of [BM18, Definition 2.5].

Example

```
gap> IsGoodMatrix(M);
#columns:[ 2, 1, 3 ], rows: [ 2, 1, 4 ]
true
```

1.1.4 ITestVector

▷ ITestVector(G , v)

(function)

G must be an FpGroup. The vector v must be orthogonal to the exponent matrix of the presentation defining G . Returns true if G satisfies the I-test for v .

Example

```
gap> ITestVector(G,[1,0,-1,2]);
# columns:[ 2, 1, 3 ], rows: [ 2, 1, 4 ]
true
```

1.1.5 ITestSimplex

▷ ITestSimplex(G)

(function)

G must be an FpGroup. Returns a vector v such that G satisfies the I-test for v or False if there is no such vector. It works by solving lots of linear programs using the package Polymaking.

Example

```
gap> F:=FreeGroup(["x","y","z"]);;
gap> AssignGeneratorVariables(F);;
gap> G:=F/[x*z*x^-1*y*z*y^-1*z^-1*y^-1, x*y^-1*x^-1*y^-1*x*y];;
gap> ITestSimplex(G);
false
```

1.1.6 ITest

▷ ITest(G , n)

(function)

G must be an FpGroup. Returns a vector v such that G satisfies the I-test for v or False if there is no such vector. First tries a number of random vectors and if none of these vectors works it uses

ITestSimplex. The number of vectors tried can be specified by the optional parameter n and by default is set to 10000.

Example

```
gap> F:=FreeGroup(8);;
gap> AssignGeneratorVariables(F);;
#I Assigned the global variables [ f1, f2, f3, f4, f5, f6, f7, f8 ]
gap> R:=[
> f6~-1*f4~-1*f7~-1*f1*f7*f5~-1*f3~-1,
> f3~-1*f1~-1*f5*f6*f8~-1*f6*f3~-1*f2,
> f1*f7~-1*f3~-1*f8~-1*f7*f2^2*f8~-1*f7~-1*f6~-1,
> f7~-1*f3*f8~-1*f1*f2*f3*f8^2*f2,
> f4~-1*f1*f2~-1*f5~-1*f2
> ];;
gap> G:=F/R;;
gap> ITest(G);
# columns:[ 1, 4, 3, 5, 2 ], rows: [ 1, 2, 3, 4, 6 ]
[ 237, 694, 243, -116, 353, -243, 1949, -162 ]
gap> F:=FreeGroup(["x","y","z"]);;
gap> AssignGeneratorVariables(F);;
gap> G:=F/[x*z*x~-1*y*z*y~-1*z~-1*y~-1, x*y~-1*x~-1*y~-1*x*y];;
gap> ITest(G);
# Tried 10000 random vectors without success.
# Now I am trying with every possible vector...
false
```

Chapter 2

Installing and Loading the Itest Package

2.1 Required Packages

The `ltest` package requires GAP (version \geq 4.9.2) and the packages `polymaking` (version \geq 0.8.1) and `semigroups` (version \geq 2.8.0)

2.2 Loading the Itest Package

To use the `ltest` Package you have to request it explicitly. This is done by calling `LoadPackage` (**Reference: LoadPackage**):

```
gap> LoadPackage("itest");
-----
Loading Itest - 1.0.1
An implementation of Barmak and Minian's I-test
by Iván Sadofski Costa (http://mate.dm.uba.ar/~isadofski)
For help, type: ?Itest
-----
true
```

If you want to load the `ltest` package by default, you can put the `LoadPackage` command into your `gaprc` file (see Section (**Reference: The gap.ini and gaprc files**)).

References

- [BM18] Jonathan Ariel Barnak and Elias Gabriel Minian. A new test for asphericity and diagrammatic reducibility of group presentations. *Proceedings of the Royal Society of Edinburgh Section A: Mathematics*, In press, 2018. [4](#), [5](#)

Index

IsGoodMatrix, [5](#)
ITest, [5](#)
Itest package, [4](#)
ITestSimplex, [5](#)
ITestVector, [5](#)

SubwordMatrix, [4](#)

WeightMatrix, [4](#)