

---

# **ilupp Documentation**

***Release 1.0.1***

**Clemens Hofreither**

**Apr 01, 2021**



---

## Contents:

---

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Preconditioners for symmetric and positive definite matrices</b> | <b>3</b>  |
| <b>2</b> | <b>Preconditioners for general matrices</b>                         | <b>5</b>  |
| <b>3</b> | <b>Stand-alone factorization functions</b>                          | <b>7</b>  |
| <b>4</b> | <b>Solving linear systems</b>                                       | <b>9</b>  |
| <b>5</b> | <b>ILU++ parameters</b>   | <b>11</b> |
| <b>6</b> | <b>Indices and tables</b>   | <b>17</b> |
|          | <b>Python Module Index</b>  | <b>19</b> |
|          | <b>Index</b>  | <b>21</b> |



The `ilupp` package provides various incomplete LU and Cholesky factorization routines for sparse matrices. It is implemented in C++ based on Jan Mayer's ILU++ package and comes with convenient Python bindings which use Scipy sparse matrices.

---

**Note:** This documentation describes only the Python bindings. The C++ interface is essentially equivalent and should be easy to figure out from the header files.

---

All preconditioner classes derive from the following base class:

**class** `ilupp._BaseWrapper` (*dtype*, *shape*)

Wrapper base class which supports methods and properties common to all preconditioners.

Implements the `scipy.sparse.linalg.LinearOperator` protocol, which means that it has a `.shape` property and can be applied to a vector using `.dot()` or simply the multiplication operator `*`.

To apply the preconditioner to a vector in place, avoiding a copy, use the `apply()` method.

**apply** (*x*)

Apply the preconditioner to the vector *x* in-place.

**factors** ()

Return all matrix factors (usually (L,U) or just (L,)) as a list of sparse matrices.

**total\_nnz**

The total number of nonzeros stored in the factor matrices of the preconditioner.



---

## Preconditioners for symmetric and positive definite matrices

---

**class** `ilupp.IChol0Preconditioner` (*A*)

An IChol(0) preconditioner (no fill-in, same sparsity pattern as *A*) for a symmetric positive definite matrix.

**Parameters** *A* – a symmetric sparse matrix in CSR or CSC format

**class** `ilupp.ICholTPreconditioner` (*A*, *add\_fill\_in*=0, *threshold*=0.0)

An incomplete Cholesky preconditioner with user-specifiable additional fill-in and threshold. With *threshold*=0, this is identical to the method described in (Lin, Moré 1999).

**Parameters**

- *A* – a symmetric sparse matrix in CSR or CSC format
- **add\_fill\_in** – the number of additional nonzeros to allow per column. By default (0), the factorization keeps the number (but not necessarily the positions) of the nonzeros identical to the original matrix.
- **threshold** – entries with a relative magnitude less than this are dropped. By default (0.0), dropping is only performed based on the number of nonzeros.





---

## Preconditioners for general matrices

---

**class** `ilupp.ILU0Preconditioner` (*A*)

An ILU(0) preconditioner (no fill-in, same sparsity pattern as *A*).

**Parameters** *A* – a symmetric sparse matrix in CSR or CSC format

**class** `ilupp.ILUTPreconditioner` (*A*, *fill\_in*=100, *threshold*=0.1)

An ILUT (incomplete LU with thresholding) preconditioner.

**Parameters**

- *A* – a sparse matrix in CSR or CSC format
- *fill\_in* – the number of nonzeros to allow per row of L/U
- *threshold* – entries with relative magnitude less than this are dropped

**class** `ilupp.ILUTPPreconditioner` (*A*, *fill\_in*=100, *threshold*=0.1, *piv\_tol*=0.1, *mem\_factor*=10.0)

An ILUTP (incomplete LU with thresholding and column pivoting) preconditioner.

**Parameters**

- *A* – a sparse matrix in CSR or CSC format
- *fill\_in* – the number of nonzeros to allow per row of L/U
- *threshold* – entries with relative magnitude less than this are dropped
- *piv\_tol* – pivoting tolerance; 0=only pivot when 0 encountered, 1=always pivot to the largest entry, inbetween: pivot depending on relative magnitude

**permutations** ()

Return a pair (L,R) of permutation arrays to be applied from the left or right due to pivoting.

**class** `ilupp.ILUCPreconditioner` (*A*, *fill\_in*=100, *threshold*=0.1)

An ILUC (Crout ILU) preconditioner. Similar to ILUT, but tends to be faster for matrices with symmetric structure. See (Li, Saad, Chow 2003).

**Parameters**

- *A* – a sparse matrix in CSR or CSC format

- **fill\_in** – the number of nonzeros to allow per column/row of L/U
- **threshold** – entries with relative magnitude less than this are dropped

**class** `ilupp.ILUCPPreconditioner` (*A, fill\_in=100, threshold=0.1, piv\_tol=0.1, mem\_factor=10.0*)  
 An ILUCP (ILUC with pivoting) preconditioner. See (Mayer 2005).

#### Parameters

- **A** – a sparse matrix in CSR or CSC format
- **fill\_in** – the number of nonzeros to allow per column/row of L/U
- **threshold** – entries with relative magnitude less than this are dropped
- **piv\_tol** – pivoting tolerance; 0=only pivot when 0 encountered, 1=always pivot to the largest entry, inbetween: pivot depending on relative magnitude

#### `permutations()`

Return a pair (L,R) of permutation arrays to be applied from the left or right due to pivoting.

**class** `ilupp.ILUppPreconditioner` (*A, threshold=1.0, fill\_in=None, params=None*)  
 A multilevel ILU++ preconditioner.

#### Parameters

- **A** – a sparse matrix in CSR or CSC format
- **fill\_in** – the fill\_in parameter for the ILU++ preconditioner
- **threshold** – the threshold parameter for ILU++; entries with relative magnitude less than this are dropped
- **params** – an instance of `iluplusplus_precond_parameter`; if passed, overrides fill\_in and threshold

---

## Stand-alone factorization functions

---

If you do not want a preconditioner object that you can apply to a vector, you can instead use these factorization functions which directly return the matrix factors of the chosen factorization. The parameters have the same meaning as for the corresponding preconditioner classes above.

Note that if you already have a preconditioner object, you can obtain the matrix factors using the `ilupp._BaseWrapper.factors()` method.

`ilupp.ichol0(A)`

Compute the L factor of an incomplete Cholesky decomposition without fill-in for the symmetric matrix A.

`ilupp.icholt(A, add_fill_in=0, threshold=0.0)`

Compute the L factor of an incomplete Cholesky decomposition with thresholding for the symmetric matrix A.

`ilupp.ilu0(A)`

Compute the (L,U) factors of an incomplete LU decomposition without fill-in.

`ilupp.ilut(A, fill_in=100, threshold=0.1)`

Compute the (L,U) factors of an incomplete LU decomposition with thresholding.

`ilupp.iluc(A, fill_in=100, threshold=0.1)`

Compute the (L,U) factors of an incomplete Crout LU decomposition with thresholding.



---

## Solving linear systems

---

The `solve()` function provides a convenient interface for setting up a preconditioner and then solving the linear system using a Krylov subspace method.

```
ilupp.solve(A, b, rtol=0.0001, atol=0.0001, max_iter=500, threshold=0.1, fill_in=None, params=None, info=False)
```

Solve the linear system  $Ax=b$  using a multilevel ILU++ preconditioner and BiCGStab.

### Parameters

- **A** – a sparse matrix in CSR or CSC format
- **b** – the right-hand side vector
- **rtol** – target relative reduction in the residual
- **atol** – target absolute magnitude of the residual
- **max\_iter** – maximum number of iterations
- **threshold** – the threshold parameter for ILU++; entries with relative magnitude less than this are dropped
- **fill\_in** – the fill\_in parameter for the ILU++ preconditioner
- **params** – an instance of `iluplusplus_precond_parameter`; if passed, overrides fill\_in and threshold
- **info** – if True, a tuple (nr\_of\_iterations, achieved\_relative\_reduction, residual\_magnitude) is returned along the solution

**Returns** a vector containing the solution  $x$



---

## ILU++ parameters

---

The class `iluplusplus_precond_parameter` provides the ability to tune advanced parameters for the multi-level ILU++ preconditioner. Some comments on these options can be found in [parameters.h](#). Further details are given in the [original publication](#).

The most important members are `iluplusplus_precond_parameter.threshold`, `iluplusplus_precond_parameter.fill_in`, and `iluplusplus_precond_parameter.piv_tol`, which have essentially the same meaning as in the functions above. Various sets of default parameters can be chosen by the `iluplusplus_precond_parameter.default_configuration()` function; see the [readme document](#) for some comments on these.

Furthermore, `iluplusplus_precond_parameter.PREPROCESSING` is an instance of `preprocessing_sequence` (see below) and can be used to choose various methods of reordering the matrix before factorization.

For example:

```
param = ilupp.iluplusplus_precond_parameter()
param.default_configuration(10)
param.PREPROCESSING.set_MAX_WEIGHTED_MATCHING_ORDERING_PQ()
```

will choose default configuration 10 and preprocess the matrix with a max weighted matching ordering followed by a PQ ordering.

**class** `ilupp.iluplusplus_precond_parameter`

**BANDWIDTH\_MULTIPLIER**

**BANDWIDTH\_OFFSET**

**BEGIN\_TOTAL\_PIV**

**COMBINE\_FACTOR**

**DROP\_TYPE\_L**

**DROP\_TYPE\_U**

EXTERNAL\_FINAL\_ROW  
EXT\_MIN\_ELIM\_FACTOR  
FINAL\_ROW\_CRIT  
FINAL\_THRESHOLD  
GLOBAL\_COMMENT  
INIT\_WEIGHTS\_LU  
MAX\_FILLIN\_IS\_INF  
MAX\_LEVELS  
MEMORY\_MAX\_LEVELS  
MEM\_FACTOR  
MIN\_ELIM\_FACTOR  
MIN\_ML\_SIZE  
MIN\_PIVOT  
MIN\_SIZE\_ZERO\_SCHUR  
MIN\_WEIGHT  
MOVE\_LEVEL\_FACTOR  
MOVE\_LEVEL\_THRESHOLD  
NEUTRAL\_ELEMENT  
PERMUTE\_ROWS  
POST\_FACT\_THRESHOLD  
PQ\_ALGORITHM  
PQ\_THRESHOLD  
PRECON\_PARAMETER  
PREPROCESSING  
REQUIRE\_ZERO\_SCHUR  
REQ\_ZERO\_SCHUR\_SIZE  
ROW\_U\_MAX  
SCALE\_WEIGHT\_INVDIAG  
SCALE\_WGT\_MAXINVDIAG  
SCHUR\_COMPLEMENT  
SIZE\_TABLE\_POS\_WEIGHTS  
SMALL\_PIVOT\_TERMINATES  
SUM\_DROPPING  
THRESHOLD\_SHIFT\_SCHUR  
THRESHOLD\_ZERO\_SCHUR  
TOTAL\_PIV



USE\_ERR\_PROP\_DROPPING  
USE\_ERR\_PROP\_DROPPING2  
USE\_FINAL\_THRESHOLD  
USE\_INVERSE\_DROPPING  
USE\_MAX\_AS\_MOVE  
USE\_PIVOT\_DROPPING  
USE\_POS\_COMPRESS  
USE\_STANDARD\_DROPPING  
USE\_STANDARD\_DROPPING2  
USE\_THRES\_ZERO\_SCHUR  
USE\_WEIGHTED\_DROPPING  
USE\_WEIGHTED\_DROPPING2  
VARIABLE\_MEM  
VARY\_THRESHOLD\_FACTOR  
WEIGHTED\_DROPPING  
WEIGHT\_ERR\_PROP\_DROP  
WEIGHT\_ERR\_PROP\_DROP2  
WEIGHT\_INVERSE\_DROP  
WEIGHT\_PIVOT\_DROP  
WEIGHT\_STANDARD\_DROP  
WEIGHT\_STANDARD\_DROP2  
WEIGHT\_TABLE\_TYPE  
WEIGHT\_WEIGHTED\_DROP  
default\_configuration (self: ilupp.ilupp.iluplusplus\_precond\_parameter, arg0: int) → None  
fill\_in  
piv\_tol  
threshold  
use\_only\_error\_propagation\_dropping1 (self: ilupp.ilupp.iluplusplus\_precond\_parameter) → None  
use\_only\_error\_propagation\_dropping2 (self: ilupp.ilupp.iluplusplus\_precond\_parameter) → None  
use\_only\_inverse\_dropping (self: ilupp.ilupp.iluplusplus\_precond\_parameter) → None  
use\_only\_pivot\_dropping (self: ilupp.ilupp.iluplusplus\_precond\_parameter) → None  
use\_only\_standard\_dropping1 (self: ilupp.ilupp.iluplusplus\_precond\_parameter) → None  
use\_only\_standard\_dropping2 (self: ilupp.ilupp.iluplusplus\_precond\_parameter) → None  
use\_only\_weighted\_dropping1 (self: ilupp.ilupp.iluplusplus\_precond\_parameter) → None  
use\_only\_weighted\_dropping2 (self: ilupp.ilupp.iluplusplus\_precond\_parameter) → None

```
class ilupp.preprocessing_sequence
```

```
    set_MAX_WEIGHTED_MATCHING_ORDERING (self: ilupp.ilupp.preprocessing_sequence) → None
    set_MAX_WEIGHTED_MATCHING_ORDERING_DD_MOV_COR_IM (self:
                                                         ilupp.ilupp.preprocessing_sequence)
                                                         → None
    set_MAX_WEIGHTED_MATCHING_ORDERING_MOVE_CORNER (self:
                                                         ilupp.ilupp.preprocessing_sequence)
                                                         → None
    set_MAX_WEIGHTED_MATCHING_ORDERING_MOVE_CORNER_IM (self:
                                                         ilupp.ilupp.preprocessing_sequence)
                                                         → None
    set_MAX_WEIGHTED_MATCHING_ORDERING_PQ (self: ilupp.ilupp.preprocessing_sequence) →
                                              None
    set_MAX_WEIGHTED_MATCHING_ORDERING_SP_MOVE_CORNER (self:
                                                         ilupp.ilupp.preprocessing_sequence)
                                                         → None
    set_MAX_WEIGHTED_MATCHING_ORDERING_SP_MOVE_CORNER_IM (self:
                                                            ilupp.ilupp.preprocessing_sequence)
                                                            → None
    set_MAX_WEIGHTED_MATCHING_ORDERING_SYMB_MOVE_CORNER (self:
                                                            ilupp.ilupp.preprocessing_sequence)
                                                            → None
    set_MAX_WEIGHTED_MATCHING_ORDERING_SYMB_MOVE_CORNER_IM (self:
                                                            ilupp.ilupp.preprocessing_sequence)
                                                            → None
    set_MAX_WEIGHTED_MATCHING_ORDERING_SYM_PQ (self: ilupp.ilupp.preprocessing_sequence)
                                                  → None
    set_MAX_WEIGHTED_MATCHING_ORDERING_UNIT_DIAG (self: ilupp.ilupp.preprocessing_sequence)
                                                    → None
    set_MAX_WEIGHTED_MATCHING_ORDERING_UNIT_DIAG_DD_MOV_COR_IM (self:
                                                                    ilupp.ilupp.preprocessing_sequence)
                                                                    → None
    set_MAX_WEIGHTED_MATCHING_ORDERING_WGT2_MOV_COR (self:
                                                         ilupp.ilupp.preprocessing_sequence)
                                                         → None
    set_MAX_WEIGHTED_MATCHING_ORDERING_WGT2_MOV_COR_IM (self:
                                                            ilupp.ilupp.preprocessing_sequence)
                                                            → None
    set_MAX_WEIGHTED_MATCHING_ORDERING_WGT_MOV_COR (self:
                                                         ilupp.ilupp.preprocessing_sequence)
                                                         → None
    set_MAX_WEIGHTED_MATCHING_ORDERING_WGT_MOV_COR_IM (self:
                                                            ilupp.ilupp.preprocessing_sequence)
                                                            → None
    set_NORM_MAX_WEIGHTED_MATCHING_ORDERING (self: ilupp.ilupp.preprocessing_sequence)
                                              → None
    set_NORM_MAX_WEIGHTED_MATCHING_ORDERING_DD_MOV_COR_IM (self:
                                                            ilupp.ilupp.preprocessing_sequence)
                                                            → None
```

```

set_NORM_MAX_WEIGHTED_MATCHING_ORDERING_MOVE_CORNER (self:
                                                    ilupp._ilupp.preprocessing_sequence)
                                                    → None
set_NORM_MAX_WEIGHTED_MATCHING_ORDERING_MOVE_CORNER_IM (self:
                                                         ilupp._ilupp.preprocessing_sequence)
                                                         → None
set_NORM_MAX_WEIGHTED_MATCHING_ORDERING_PQ (self: ilupp._ilupp.preprocessing_sequence)
                                                    → None
set_NORM_MAX_WEIGHTED_MATCHING_ORDERING_SP_MOVE_CORNER (self:
                                                         ilupp._ilupp.preprocessing_sequence)
                                                         → None
set_NORM_MAX_WEIGHTED_MATCHING_ORDERING_SP_MOVE_CORNER_IM (self:
                                                            ilupp._ilupp.preprocessing_sequence)
                                                            → None
set_NORM_MAX_WEIGHTED_MATCHING_ORDERING_SYMB_MOVE_CORNER (self:
                                                            ilupp._ilupp.preprocessing_sequence)
                                                            → None
set_NORM_MAX_WEIGHTED_MATCHING_ORDERING_SYMB_MOVE_CORNER_IM (self:
                                                                ilupp._ilupp.preprocessing_sequence)
                                                                → None
set_NORM_MAX_WEIGHTED_MATCHING_ORDERING_SYM_PQ (self:
                                                  ilupp._ilupp.preprocessing_sequence)
                                                  → None
set_NORM_MAX_WEIGHTED_MATCHING_ORDERING_WGT2_MOV_COR (self:
                                                         ilupp._ilupp.preprocessing_sequence)
                                                         → None
set_NORM_MAX_WEIGHTED_MATCHING_ORDERING_WGT2_MOV_COR_IM (self:
                                                            ilupp._ilupp.preprocessing_sequence)
                                                            → None
set_NORM_MAX_WEIGHTED_MATCHING_ORDERING_WGT_MOV_COR (self:
                                                         ilupp._ilupp.preprocessing_sequence)
                                                         → None
set_NORM_MAX_WEIGHTED_MATCHING_ORDERING_WGT_MOV_COR_IM (self:
                                                            ilupp._ilupp.preprocessing_sequence)
                                                            → None
set_PQ (self: ilupp._ilupp.preprocessing_sequence) → None
set_SPARSE_FIRST (self: ilupp._ilupp.preprocessing_sequence) → None
set_SPARSE_FIRST_MAX_WEIGHTED_MATCHING_ORDERING (self:
                                                    ilupp._ilupp.preprocessing_sequence)
                                                    → None
set_SPARSE_FIRST_MAX_WEIGHTED_MATCHING_ORDERING_DD_MOV_COR_IM (self:
                                                                ilupp._ilupp.preprocessing_sequence)
                                                                → None
set_SPARSE_FIRST_MAX_WEIGHTED_MATCHING_ORDERING_UNIT_DIAG (self:
                                                            ilupp._ilupp.preprocessing_sequence)
                                                            → None
set_SPARSE_FIRST_MAX_WEIGHTED_MATCHING_ORDERING_UNIT_DIAG_DD_MOV_COR_IM (self:
                                                                                ilupp._ilupp.preprocessing_sequence)
                                                                                →
                                                                                None

```

**set\_none** (*self*: *ilupp.\_ilupp.preprocessing\_sequence*) → None

**set\_normalize** (*self*: *ilupp.\_ilupp.preprocessing\_sequence*) → None

**to\_names** (*self*: *ilupp.\_ilupp.preprocessing\_sequence*) → list

## CHAPTER 6

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`



**i**

ilupp, [1](#)





## Symbols

`_BaseWrapper` (class in `ilupp`), 1

## A

`apply()` (`ilupp._BaseWrapper` method), 1

## B

`BANDWIDTH_MULTIPLIER`  
(`ilupp.iluplusplus_precond_parameter` attribute), 11

`BANDWIDTH_OFFSET` (`ilupp.iluplusplus_precond_parameter` attribute), 11

`BEGIN_TOTAL_PIV` (`ilupp.iluplusplus_precond_parameter` attribute), 11

## C

`COMBINE_FACTOR` (`ilupp.iluplusplus_precond_parameter` attribute), 11

## D

`default_configuration()`  
(`ilupp.iluplusplus_precond_parameter` method), 13

`DROP_TYPE_L` (`ilupp.iluplusplus_precond_parameter` attribute), 11

`DROP_TYPE_U` (`ilupp.iluplusplus_precond_parameter` attribute), 11

## E

`EXT_MIN_ELIM_FACTOR`  
(`ilupp.iluplusplus_precond_parameter` attribute), 12

`EXTERNAL_FINAL_ROW`  
(`ilupp.iluplusplus_precond_parameter` attribute), 11

## F

`factors()` (`ilupp._BaseWrapper` method), 1

`fill_in` (`ilupp.iluplusplus_precond_parameter` attribute), 13

`FINAL_ROW_CRIT` (`ilupp.iluplusplus_precond_parameter` attribute), 12

`FINAL_THRESHOLD` (`ilupp.iluplusplus_precond_parameter` attribute), 12

## G

`GLOBAL_COMMENT` (`ilupp.iluplusplus_precond_parameter` attribute), 12

`ichol0()` (in module `ilupp`), 7

`ICHol0Preconditioner` (class in `ilupp`), 3

`icholt()` (in module `ilupp`), 7

`ICHoltPreconditioner` (class in `ilupp`), 3

`ilu0()` (in module `ilupp`), 7

`ILU0Preconditioner` (class in `ilupp`), 5

`iluc()` (in module `ilupp`), 7

`ILUCPPreconditioner` (class in `ilupp`), 6

`ILUCPreconditioner` (class in `ilupp`), 5

`iluplusplus_precond_parameter` (class in `ilupp`), 11

`ilupp` (module), 1

`ILUpPreconditioner` (class in `ilupp`), 6

`ilut()` (in module `ilupp`), 7

`ILUTPPreconditioner` (class in `ilupp`), 5

`ILUTPreconditioner` (class in `ilupp`), 5

`INIT_WEIGHTS_LU` (`ilupp.iluplusplus_precond_parameter` attribute), 12

## M

`MAX_FILLIN_IS_INF`  
(`ilupp.iluplusplus_precond_parameter` attribute), 12

`MAX_LEVELS` (`ilupp.iluplusplus_precond_parameter` attribute), 12

`MEM_FACTOR` (`ilupp.iluplusplus_precond_parameter` attribute), 12

MEMORY\_MAX\_LEVELS  
(*ilupp.iluplusplus\_precond\_parameter attribute*), 12

MIN\_ELIM\_FACTOR (*ilupp.iluplusplus\_precond\_parameter attribute*), 12

MIN\_ML\_SIZE (*ilupp.iluplusplus\_precond\_parameter attribute*), 12

MIN\_PIVOT (*ilupp.iluplusplus\_precond\_parameter attribute*), 12

MIN\_SIZE\_ZERO\_SCHUR  
(*ilupp.iluplusplus\_precond\_parameter attribute*), 12

MIN\_WEIGHT (*ilupp.iluplusplus\_precond\_parameter attribute*), 12

MOVE\_LEVEL\_FACTOR  
(*ilupp.iluplusplus\_precond\_parameter attribute*), 12

MOVE\_LEVEL\_THRESHOLD  
(*ilupp.iluplusplus\_precond\_parameter attribute*), 12

## N

NEUTRAL\_ELEMENT (*ilupp.iluplusplus\_precond\_parameter attribute*), 12

## P

permutations() (*ilupp.ILUCPPreconditioner method*), 6

permutations() (*ilupp.ILUTPPreconditioner method*), 5

PERMUTE\_ROWS (*ilupp.iluplusplus\_precond\_parameter attribute*), 12

piv\_tol (*ilupp.iluplusplus\_precond\_parameter attribute*), 13

POST\_FACT\_THRESHOLD  
(*ilupp.iluplusplus\_precond\_parameter attribute*), 12

PQ\_ALGORITHM (*ilupp.iluplusplus\_precond\_parameter attribute*), 12

PQ\_THRESHOLD (*ilupp.iluplusplus\_precond\_parameter attribute*), 12

PRECON\_PARAMETER (*ilupp.iluplusplus\_precond\_parameter attribute*), 12

PREPROCESSING (*ilupp.iluplusplus\_precond\_parameter attribute*), 12

preprocessing\_sequence (class in *ilupp*), 13

## R

REQ\_ZERO\_SCHUR\_SIZE  
(*ilupp.iluplusplus\_precond\_parameter attribute*), 12

REQUIRE\_ZERO\_SCHUR  
(*ilupp.iluplusplus\_precond\_parameter attribute*), 12

ROW\_U\_MAX (*ilupp.iluplusplus\_precond\_parameter attribute*), 12

## S

SCALE\_WEIGHT\_INVDIAG  
(*ilupp.iluplusplus\_precond\_parameter attribute*), 12

SCALE\_WGT\_MAXINVDIAG  
(*ilupp.iluplusplus\_precond\_parameter attribute*), 12

SCHUR\_COMPLEMENT (*ilupp.iluplusplus\_precond\_parameter attribute*), 12

set\_MAX\_WEIGHTED\_MATCHING\_ORDERING()  
(*ilupp.preprocessing\_sequence method*), 14

set\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_DD\_MOV\_COR\_IM()  
(*ilupp.preprocessing\_sequence method*), 14

set\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_MOVE\_CORNER()  
(*ilupp.preprocessing\_sequence method*), 14

set\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_MOVE\_CORNER\_IM()  
(*ilupp.preprocessing\_sequence method*), 14

set\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_PQ()  
(*ilupp.preprocessing\_sequence method*), 14

set\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_SP\_MOVE\_CORNER()  
(*ilupp.preprocessing\_sequence method*), 14

set\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_SP\_MOVE\_CORNER\_I()  
(*ilupp.preprocessing\_sequence method*), 14

set\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_SYM\_PQ()  
(*ilupp.preprocessing\_sequence method*), 14

set\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_SYMB\_MOVE\_CORNER()  
(*ilupp.preprocessing\_sequence method*), 14

set\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_SYMB\_MOVE\_CORNER\_I()  
(*ilupp.preprocessing\_sequence method*), 14

set\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_UNIT\_DIAG()  
(*ilupp.preprocessing\_sequence method*), 14

set\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_UNIT\_DIAG\_DD\_MOV()  
(*ilupp.preprocessing\_sequence method*), 14

set\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_WGT2\_MOV\_COR()  
(*ilupp.preprocessing\_sequence method*), 14

set\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_WGT2\_MOV\_COR\_IM()  
(*ilupp.preprocessing\_sequence method*), 14

set\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_WGT\_MOV\_COR()  
(*ilupp.preprocessing\_sequence method*), 14

set\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_WGT\_MOV\_COR\_IM()  
(*ilupp.preprocessing\_sequence method*), 14

set\_none() (*ilupp.preprocessing\_sequence method*), 15

set\_NORM\_MAX\_WEIGHTED\_MATCHING\_ORDERING()  
(*ilupp.preprocessing\_sequence method*), 14

set\_NORM\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_DD\_MOV\_COR()  
(*ilupp.preprocessing\_sequence method*), 14

set\_NORM\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_MOVE\_CORNER()  
(*ilupp.preprocessing\_sequence method*), 14

set\_NORM\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_MOVE\_CORNER\_I()  
(*ilupp.preprocessing\_sequence method*), 15

set\_NORM\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_PQ (names () (ilupp.preprocessing\_sequence method), 15  
 (ilupp.preprocessing\_sequence method), 16  
 set\_NORM\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_SP\_MOVE\_COR\_IM (ilupp.iluplusplus\_precond\_parameter at-  
 (ilupp.preprocessing\_sequence method), 15 TOTAL\_PIV (ilupp.iluplusplus\_precond\_parameter at-  
 set\_NORM\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_SP\_MOVE\_COR\_IM (ilupp.iluplusplus\_precond\_parameter at-  
 (ilupp.preprocessing\_sequence method), 15  
 set\_NORM\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_SYM\_MOVE\_COR\_IM (ilupp.iluplusplus\_precond\_parameter at-  
 (ilupp.preprocessing\_sequence method), 15 USE\_ERR\_PROP\_DROPPING  
 set\_NORM\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_SYM\_MOVE\_COR\_IM (ilupp.iluplusplus\_precond\_parameter at-  
 (ilupp.preprocessing\_sequence method), 15 tribute), 12  
 set\_NORM\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_SYM\_MOVE\_COR\_IM (ilupp.iluplusplus\_precond\_parameter at-  
 (ilupp.preprocessing\_sequence method), 15 tribute), 12  
 set\_NORM\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_WGT2\_MOVE\_COR\_IM (ilupp.iluplusplus\_precond\_parameter at-  
 (ilupp.preprocessing\_sequence method), 15 USE\_FINAL\_THRESHOLD  
 set\_NORM\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_WGT2\_MOVE\_COR\_IM (ilupp.iluplusplus\_precond\_parameter at-  
 (ilupp.preprocessing\_sequence method), 15 tribute), 13  
 set\_NORM\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_WGT2\_MOVE\_COR\_IM (ilupp.iluplusplus\_precond\_parameter at-  
 (ilupp.preprocessing\_sequence method), 15 USE\_MAX\_AS\_MOVE (ilupp.iluplusplus\_precond\_parameter at-  
 set\_normalize () (ilupp.preprocessing\_sequence method), 16 use\_only\_error\_propagation\_dropping1 ()  
 set\_PQ () (ilupp.preprocessing\_sequence method), 15 (ilupp.iluplusplus\_precond\_parameter  
 set\_SPARSE\_FIRST () (ilupp.preprocessing\_sequence method), 13  
 (ilupp.preprocessing\_sequence method), use\_only\_error\_propagation\_dropping2 ()  
 15 (ilupp.iluplusplus\_precond\_parameter  
 set\_SPARSE\_FIRST\_MAX\_WEIGHTED\_MATCHING\_ORDERING (ilupp.preprocessing\_sequence method), 13  
 (ilupp.preprocessing\_sequence method), 15 use\_only\_inverse\_dropping ()  
 set\_SPARSE\_FIRST\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_DD\_MOVE\_COR\_IM (ilupp.iluplusplus\_precond\_parameter  
 (ilupp.preprocessing\_sequence method), 15 method), 13  
 set\_SPARSE\_FIRST\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_UNIT\_DIAG () use\_only\_pivot\_dropping ()  
 (ilupp.preprocessing\_sequence method), 15 (ilupp.iluplusplus\_precond\_parameter  
 set\_SPARSE\_FIRST\_MAX\_WEIGHTED\_MATCHING\_ORDERING\_INIT\_DIAG\_DD\_MOVE\_COR\_IM (ilupp.iluplusplus\_precond\_parameter  
 (ilupp.preprocessing\_sequence method), 15 method), 13  
 use\_only\_standard\_dropping1 ()  
 SIZE\_TABLE\_POS\_WEIGHTS (ilupp.iluplusplus\_precond\_parameter at-  
 (ilupp.iluplusplus\_precond\_parameter at- method), 13  
 use\_only\_standard\_dropping2 ()  
 SMALL\_PIVOT\_TERMINATES (ilupp.iluplusplus\_precond\_parameter at-  
 (ilupp.iluplusplus\_precond\_parameter at- method), 13  
 use\_only\_weighted\_dropping1 ()  
 solve () (in module ilupp), 9 (ilupp.iluplusplus\_precond\_parameter  
 SUM\_DROPPING (ilupp.iluplusplus\_precond\_parameter method), 13  
 attribute), 12 use\_only\_weighted\_dropping2 ()  
 (ilupp.iluplusplus\_precond\_parameter  
 method), 13  
**T**  
 threshold (ilupp.iluplusplus\_precond\_parameter at-  
 (ilupp.iluplusplus\_precond\_parameter at- USE\_PIVOT\_DROPPING  
 tribute), 13 (ilupp.iluplusplus\_precond\_parameter at-  
 THRESHOLD\_SHIFT\_SCHUR (ilupp.iluplusplus\_precond\_parameter at-  
 (ilupp.iluplusplus\_precond\_parameter at- USE\_POS\_COMPRESS (ilupp.iluplusplus\_precond\_parameter  
 tribute), 12 attribute), 13  
 THRESHOLD\_ZERO\_SCHUR USE\_STANDARD\_DROPPING  
 (ilupp.iluplusplus\_precond\_parameter at-  
 (ilupp.iluplusplus\_precond\_parameter at- tribute), 13  
 USE\_STANDARD\_DROPPING2

*(ilupp.iluplusplus\_precond\_parameter  
attribute), 13*  
USE\_THRES\_ZERO\_SCHUR  
*(ilupp.iluplusplus\_precond\_parameter  
attribute), 13*  
USE\_WEIGHTED\_DROPPING  
*(ilupp.iluplusplus\_precond\_parameter  
attribute), 13*  
USE\_WEIGHTED\_DROPPING2  
*(ilupp.iluplusplus\_precond\_parameter  
attribute), 13*

## V

VARIABLE\_MEM *(ilupp.iluplusplus\_precond\_parameter  
attribute), 13*  
VARY\_THRESHOLD\_FACTOR  
*(ilupp.iluplusplus\_precond\_parameter  
attribute), 13*

## W

WEIGHT\_ERR\_PROP\_DROP  
*(ilupp.iluplusplus\_precond\_parameter  
attribute), 13*  
WEIGHT\_ERR\_PROP\_DROP2  
*(ilupp.iluplusplus\_precond\_parameter  
attribute), 13*  
WEIGHT\_INVERSE\_DROP  
*(ilupp.iluplusplus\_precond\_parameter  
attribute), 13*  
WEIGHT\_PIVOT\_DROP  
*(ilupp.iluplusplus\_precond\_parameter  
attribute), 13*  
WEIGHT\_STANDARD\_DROP  
*(ilupp.iluplusplus\_precond\_parameter  
attribute), 13*  
WEIGHT\_STANDARD\_DROP2  
*(ilupp.iluplusplus\_precond\_parameter  
attribute), 13*  
WEIGHT\_TABLE\_TYPE  
*(ilupp.iluplusplus\_precond\_parameter  
attribute), 13*  
WEIGHT\_WEIGHTED\_DROP  
*(ilupp.iluplusplus\_precond\_parameter  
attribute), 13*  
WEIGHTED\_DROPPING  
*(ilupp.iluplusplus\_precond\_parameter  
attribute), 13*