NAG Library Function Document

nag_opt_sparse_convex_qp_option_set_double (e04nuc)

1 Purpose

nag_opt_sparse_convex_qp_option_set_double (e04nuc) may be used to supply individual double optional parameters to nag_opt_sparse_convex_qp_solve (e04nqc). The initialization function nag_opt_sparse_convex_qp_init (e04npc) **must** have been called before calling nag_opt_sparse_convex_qp_option_set_double (e04nuc).

2 Specification

```
#include <nag.h>
#include <nage04.h>
void nag_opt_sparse_convex_qp_option_set_double (const char *string,
     double rvalue, Nag_E04State *state, NagError *fail)
```

3 Description

nag_opt_sparse_convex_qp_option_set_double (e04nuc) may be used to supply values for double optional parameters to nag_opt_sparse_convex_qp_solve (e04nqc). It is only necessary to call nag_opt_sparse_convex_qp_option_set_double (e04nuc) for those arguments whose values are to be different from their default values. One call to nag_opt_sparse_convex_qp_option_set_double (e04nuc) sets one argument value.

Each double optional parameter is defined by a single character string in **string** and the corresponding value in **rvalue**. For example the following illustrates how the *LU* stability tolerance could be defined:

```
factol = 100.0;
 if (illcon) factol = 5.0;
 e04nuc ("LU Factor Tolerance", factol, &state, &fail);
```

Optional parameter settings are preserved following a call to nag_opt_sparse_convex_qp_solve (e04nqc) and so the keyword **Defaults** is provided to allow you to reset all the optional parameters to their default values before a subsequent call to nag_opt_sparse_convex_qp_solve (e04nqc).

A complete list of optional parameters, their abbreviations, synonyms and default values is given in Section 12 in nag_opt_sparse_convex_qp_solve (e04nqc).

4 References

None.

5 Arguments

1: **string** – const char * *Input*

On entry: a single valid keyword of a double optional parameter (as described in Section 12 in nag_opt_sparse_convex_qp_solve (e04nqc)).

2: **rvalue** – double *Input*

On entry: the value associated with the keyword in **string**.

3: **state** – Nag_E04State * *Communication Structure*

state contains internal information required for functions in this suite. It must not be modified in any way.

4: **fail** – NagError * *Input/Output*

The NAG error argument (see Section 2.7 in How to Use the NAG Library and its Documentation).

6 Error Indicators and Warnings

NE_ALLOC_FAIL

Dynamic memory allocation failed.
See Section 3.2.1.2 in How to Use the NAG Library and its Documentation for further information.

NE_BAD_PARAM

On entry, argument ⟨*value*⟩ had an illegal value.

NE_E04_OPTION_INVALID

The supplied option is invalid. Check that the keywords are neither ambiguous nor misspelt. The option string is '⟨*value*⟩' and **rvalue** = ⟨*value*⟩.

NE_E04NPC_NOT_INIT

The initialization function nag_opt_sparse_convex_qp_init (e04npc) has not been called.

NE_INTERNAL_ERROR

An internal error has occurred in this function. Check the function call and any array sizes. If the call is correct then please contact NAG for assistance.

An unexpected error has been triggered by this function. Please contact NAG.
See Section 3.6.6 in How to Use the NAG Library and its Documentation for further information.

NE_NO_LICENCE

Your licence key may have expired or may not have been installed correctly.
See Section 3.6.5 in How to Use the NAG Library and its Documentation for further information.

7 Accuracy

Not applicable.

8 Parallelism and Performance

nag_opt_sparse_convex_qp_option_set_double (e04nuc) is not threaded in any implementation.

9 Further Comments

nag_opt_sparse_convex_qp_option_set_file (e04nrc) or nag_opt_sparse_convex_qp_option_set_string (e04nsc) may also be used to supply double optional parameters to nag_opt_sparse_convex_qp_solve (e04nqc).

10 Example

See Section 10 in nag_opt_sparse_convex_qp_option_set_file (e04nrc).