

NAG Library Function Document

nag_opt_sparse_convex_qp_option_set_string (e04nsc)

1 Purpose

nag_opt_sparse_convex_qp_option_set_string (e04nsc) may be used to supply individual optional arguments 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_string (e04nsc).

2 Specification

```
#include <nag.h>
#include <nage04.h>

void nag_opt_sparse_convex_qp_option_set_string (const char *string,
        Nag_E04State *state, NagError *fail)
```

3 Description

nag_opt_sparse_convex_qp_option_set_string (e04nsc) may be used to supply values for optional arguments to nag_opt_sparse_convex_qp_solve (e04nqc). It is only necessary to call nag_opt_sparse_convex_qp_option_set_string (e04nsc) for those arguments whose values are to be different from their default values. One call to nag_opt_sparse_convex_qp_option_set_string (e04nsc) sets one argument value.

Each optional argument is defined by a single character string, consisting of one or more items. The items associated with a given option must be separated by spaces, or equals signs [=]. Alphabetic characters may be upper or lower case. The string

```
Print Level = 1
```

is an example of a string used to set an optional argument. For each option the string contains one or more of the following items:

- a mandatory keyword;
- a phrase that qualifies the keyword;
- a number that specifies an Integer or double value. Such numbers may be up to 16 contiguous characters which can be read using C's d or g formats, terminated by a space if this is not the last item on the line.

For nag_opt_sparse_convex_qp_option_set_string (e04nsc), each user-specified option is not normally printed as it is defined, but this printing may be turned on using the keyword **List**. Thus the statement

```
e04nsc ("List", &state, &fail);
```

turns on printing of this and subsequent options. Printing may be turned off again using the keyword **Nolist**.

Optional argument 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 arguments to their default values before a subsequent call to nag_opt_sparse_convex_qp_solve (e04nqc).

A complete list of optional arguments, 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 option string (see Section 3 in nag_opt_sparse_convex_qp_option_set_string (e04nsc) and Section 12 in nag_opt_sparse_convex_qp_solve (e04nqc)).
- 2: **state** – Nag_E04State * *Communication Structure*
state contains internal information required for functions in this suite. It must not be modified in any way.
- 3: **fail** – NagError * *Input/Output*
 The NAG error argument (see Section 3.6 in the Essential Introduction).

6 Error Indicators and Warnings

NE_BAD_PARAM

On entry, argument $\langle value \rangle$ 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 ' $\langle value \rangle$ '.

NE_E04NPC_NOT_INIT

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.

7 Accuracy

Not applicable.

8 Parallelism and Performance

Not applicable.

9 Further Comments

nag_opt_sparse_convex_qp_option_set_file (e04nrc), nag_opt_sparse_convex_qp_option_set_integer (e04ntc) or nag_opt_sparse_convex_qp_option_set_double (e04nuc) may also be used to supply optional arguments to nag_opt_sparse_convex_qp_solve (e04nqc).

10 Example

See Section 10 in nag_opt_sparse_convex_qp_option_set_file (e04nrc).
