QDAGI

Integrates a function over an infinite or semi-infinite interval.

Required Arguments

F — User-supplied `FUNCTION` to be integrated. The form is
`F(X)`, where

`X` – Independent variable. (Input)
`F` – The function value. (Output)

`F` must be declared `EXTERNAL` in the calling program.

BOUND — Finite bound of the integration range. (Input)
Ignored if `INTERV` = 2.

INTERV — Flag indicating integration interval. (Input)

INTERV	Interval
-1	$(-\infty$, `BOUND`$)$
1	$($`BOUND`$, +\infty)$
2	$(-\infty, +\infty)$

RESULT — Estimate of the integral from `A` to `B` of `F`. (Output)

Optional Arguments

ERRABS — Absolute accuracy desired. (Input)
Default: `ERRABS` = 1.e-3 for single precision and 1.d-8 for double precision.

ERRREL — Relative accuracy desired. (Input)
Default: `ERRREL` = 1.e-3 for single precision and 1.d-8 for double precision.

ERREST — Estimate of the absolute value of the error. (Output)

FORTRAN 90 Interface

Generic: `CALL QDAGI (F, BOUND, INTERV, RESULT [,…])`

Specific: The specific interface names are `S_QDAGI` and `D_QDAGI`.

FORTRAN 77 Interface

Single: `CALL QDAGI (F, BOUND, INTERV, ERRABS, ERRREL, RESULT, ERREST)`

Double: The double precision name is `DQDAGI`.

Description

The routine `QDAGI` uses a globally adaptive scheme in an attempt to reduce the absolute error. It initially transforms an infinite or semi-infinite interval into the finite interval [0, 1]. Then, `QDAGI` uses a 21-point Gauss-Kronrod rule to estimate the integral and the err It bisects any interval with an unacceptable error estimate and continues this process until termination. This routine is designed to handl endpoint singularities. In addition to the general strategy described in QDAG, this subroutine employs an extrapolation procedure known the ε-algorithm. The routine `QDAGI` is an implementation of the subroutine `QAGI`, which is fully documented by Piessens et al. (1983).

Comments

1. Workspace may be explicitly provided, if desired, by use of `Q2AGI/DQ2AGI`. The reference is

```
CALL Q2AGI (F, BOUND, INTERV, ERRABS, ERRREL, RESULT, ERREST, MAXSUB, NEVAL,
NSUBIN, ALIST, BLIST, RLIST, ELIST, IORD)
```

The additional arguments are as follows:

MAXSUB — Number of subintervals allowed. (Input)
A value of 500 is used by `QDAGI`.

NEVAL — Number of evaluations of `F`. (Output)

NSUBIN — Number of subintervals generated. (Output)

ALIST — Array of length `MAXSUB` containing a list of the `NSUBIN` left endpoints. (Output)

BLIST — Array of length `MAXSUB` containing a list of the `NSUBIN` right endpoints. (Output)

***RLIST* —** Array of length MAXSUB containing approximations to the NSUBIN integrals over the intervals defined by ALIST, BLIST. (Output)

ELIST — Array of length MAXSUB containing the error estimates of the NSUBIN values in RLIST. (Output)

IORD — Array of length MAXSUB. (Output)
Let K be NSUBIN if NSUBIN .LE.(MAXSUB/2 + 2), MAXSUB + 1 − NSUBIN otherwise. The first K locations contain pointers to the error estimates over the subintervals, such that ELIST(IORD(1)), …, ELIST(IORD(K)) form a decreasing sequence.

2. Informational errors

Type	Code	
4	1	The maximum number of subintervals allowed has been reached.
3	2	Roundoff error, preventing the requested tolerance from being achieved, has been detected.
3	3	A degradation in precision has been detected.
3	4	Roundoff error in the extrapolation table, preventing the requested tolerance from being achieved, has been detected.
4	5	Integral is divergent or slowly convergent.

3. If EXACT is the exact value, QDAGI attempts to find RESULT such that ABS(EXACT − RESULT).LE.MAX(ERRABS, ERRREL * ABS(EXACT)). To specify only a relative error, set ERRABS to zero. Similarly, to specify only an absolute error, set ERRREL to zero.

Example

The value of

$$\int_0^\infty \frac{\ln(x)}{1+(10x)^2}\,dx = \frac{-\pi \ln(10)}{20}$$

is estimated. The values of the actual and estimated error are machine dependent. Note that we have requested an absolute error of 0 an relative error of .001. The effect of these requests, as documented in Comment 3 above, is to ignore the absolute error requirement.

```
      USE QDAGI_INT
      USE UMACH_INT
      USE CONST_INT

      IMPLICIT   NONE
      INTEGER    INTERV, NOUT
      REAL       ABS, ALOG, BOUND, ERRABS, ERREST, ERROR, &
                ERRREL, EXACT, F, PI, RESULT
      INTRINSIC  ABS, ALOG
      EXTERNAL   F
!                                 Get output unit number
      CALL UMACH (2, NOUT)
!                                 Set limits of integration
      BOUND  = 0.0
      INTERV = 1
!                                 Set error tolerances
      ERRABS = 0.0
      CALL QDAGI (F, BOUND, INTERV, RESULT, ERRABS=ERRABS,  &
                 ERREST=ERREST)
!                                 Print results
      PI    = CONST('PI')
      EXACT = -PI*ALOG(10.)/20.
      ERROR = ABS(RESULT-EXACT)
      WRITE (NOUT,99999) RESULT, EXACT, ERREST, ERROR
99999 FORMAT (' Computed =', F8.3, 13X, ' Exact =', F8.3//' Error ', &
            'estimate =', 1PE10.3, 6X, 'Error =', 1PE10.3)
      END
!
      REAL FUNCTION F (X)
      REAL       X
      REAL       ALOG
      INTRINSIC  ALOG
      F = ALOG(X)/(1.+(10.*X)**2)
      RETURN
      END
```

Output

```
Computed =  -0.362              Exact =  -0.362

Error estimate = 2.652E-06      Error = 5.960E-08
```

Visual Numerics, Inc.
Visual Numerics - Developers of IMSL and PV-WAVE
http://www.vni.com/
PHONE: 713.784.3131
FAX:713.781.9260