A Definite Integration Interface for REDUCE

Kerry Gaskell
Konrad–Zuse–Zentrum für Informationstechnik Berlin
Takustraše 7
D–14195 Berlin – Dahlem
Federal Republic of Germany
E–mail: neun@zib.de*

July 1994

1 Introduction

This documentation describes part of REDUCE's definite integration package that is able to calculate the definite integrals of many functions, including several special functions. There are other parts of this package, such as Stan Kameny's code for contour integration, that are not included here. The integration process described here is not the more normal approach of initially calculating the indefinite integral, but is instead the rather unusual idea of representing each function as a Meijer G-function (a formal definition of the Meijer G-function can be found in [1]), and then calculating the integral by using the following Meijer G integration formula.

$$\int_0^\infty x^{\alpha-1} G^{st}_{uv}\left(\sigma x \;\middle|\; \begin{matrix}(c_u)\\(d_v)\end{matrix}\right) G^{mn}_{pq}\left(\omega x^{l/k} \;\middle|\; \begin{matrix}(a_p)\\(b_q)\end{matrix}\right) dx = k G^{ij}_{kl}\left(\xi \;\middle|\; \begin{matrix}(g_k)\\(h_l)\end{matrix}\right) \qquad (1)$$

The resulting Meijer G-function is then retransformed, either directly or via a hypergeometric function simplification, to give the answer. A more detailed account of this theory can be found in [2].

[1]This definite integration interface was written during my one year placement at ZIB. Any comments and/or problems should therefore be directed to Winfried Neun at neun@zib.de.

2 Integration between zero and infinity

As an example, if one wishes to calculate the following integral

$$\int_0^\infty x^{-1} e^{-x} sin(x)\, dx$$

then initially the correct Meijer G-functions are found, via a pattern matching process, and are substituted into (1) to give

$$\sqrt{\pi} \int_0^\infty x^{-1} G_{01}^{10}\left(x \,\middle|\, \begin{matrix} \cdot \\ 0 \end{matrix}\right) G_{02}^{10}\left(\frac{x^2}{4} \,\middle|\, \begin{matrix} \cdot\ \cdot \\ \frac{1}{2}\ 0 \end{matrix}\right) dx$$

The cases for validity of the integral are then checked. If these are found to be satisfactory then the formula is calculated and we obtain the following Meijer G-function

$$G_{22}^{12}\left(1 \,\middle|\, \begin{matrix} \frac{1}{2}\ 1 \\ \frac{1}{2}\ 0 \end{matrix}\right)$$

This is reduced to the following hypergeometric function

$${}_2F_1(\tfrac{1}{2}, 1; \tfrac{3}{2}; -1)$$

which is then calculated to give the correct answer of

$$\frac{\pi}{4}$$

The above formula (1) is also true for the integration of a single Meijer G-function by replacing the second Meijer G-function with a trivial Meijer G-function.

A list of numerous particular Meijer G-functions is available in [1].

3 Integration over other ranges

Although the description so far has been limited to the computation of definite integrals between 0 and infinity, it can also be extended to calculate integrals between 0 and some specific upper bound, and by further extension, integrals between any two bounds. One approach is to use the Heaviside function, i.e.

$$\int_0^\infty x^2 e^{-x} H(1-x)\,dx = \int_0^1 x^2 e^{-x} dx$$

Another approach, again not involving the normal indefinite integration process, again uses Meijer G-functions, this time by means of the following formula

$$\int_0^y x^{\alpha-1} G_{pq}^{mn}\left(\sigma x \,\middle|\, \begin{matrix}(a_u)\\(b_v)\end{matrix}\right) dx = y^\alpha\, G_{p+1\;q+1}^{m\;n+1}\left(\sigma y \,\middle|\, \begin{matrix}(a_1..a_n, 1-\alpha, a_{n+1}..a_p)\\(b_1..b_m, -\alpha, b_{m+1}..b_q)\end{matrix}\right) \quad (2)$$

For a more detailed look at the theory behind this see [2].

For example, if one wishes to calculate the following integral

$$\int_0^y sin(2\sqrt{x})\,dx$$

then initially the correct Meijer G-function is found, by a pattern matching process, and is substituted into (2) to give

$$\int_0^y G_{02}^{10}\left(x \,\middle|\, \begin{matrix}.\;\;.\\ \frac{1}{2}\;0\end{matrix}\right) dx$$

which then in turn gives

$$y\; G_{13}^{11}\left(y \,\middle|\, \begin{matrix}0\\ \frac{1}{2}\;-1\;0\end{matrix}\right) dx$$

and returns the result

$$\frac{\sqrt{\pi}\, J_{3/2}(2\,\sqrt{y})\, y}{y^{1/4}}$$

4 Using the definite integration package

To use this package, you must first load it by the command

```
load_package defint;
```

Definite integration is then possible using the `int` command with the syntax:

```
INT(EXPRN:algebraic,VAR:kernel,LOW:algebraic,UP:algebraic)
    :algebraic.
```

where LOW and UP are the lower and upper bounds respectively for the definite integration of EXPRN with respect to VAR.

4.1 Examples

$$\int_0^\infty e^{-x} dx$$

```
int(e^(-x),x,0,infinity);

1
```

$$\int_0^\infty x sin(1/x)\, dx$$

```
int(x*sin(1/x),x,0,infinity);

           1
INT(X*SIN(---),X,0,INFINITY)
           X
```

$$\int_0^\infty x^2 cos(x)\, e^{-2x} dx$$

```
int(x^2*cos(x)*e^(-2*x),x,0,infinity);

  4
-----
 125
```

$$\int_0^\infty x e^{-1/2x} H(1-x)\, dx = \int_0^1 x e^{-1/2x} dx$$

```
int(x*e^(-1/2x)*Heaviside(1-x),x,0,infinity);

 2*(2*SQRT(E) - 3)
-------------------
      SQRT(E)
```

$$\int_0^1 x\,log(1+x)\,dx$$

```
int(x*log(1+x),x,0,1);

 1
---
 4
```

$$\int_0^y cos(2x)\,dx$$

```
int(cos(2x),x,y,2y);

 SIN(4*Y) - SIN(2*Y)
---------------------
          2
```

5 Integral Transforms

A useful application of the definite integration package is in the calculation of various integral transforms. The transforms available are as follows:

- Laplace transform
- Hankel transform
- Y-transform
- K-transform
- StruveH transform
- Fourier sine transform
- Fourier cosine transform

5.1 Laplace transform

The Laplace transform

$$f(s) = \mathcal{L}\ \{\mathrm{F(t)}\} = \int_0^\infty e^{-st} F(t)\, dt$$

can be calculated by using the `laplace_transform` command.

This requires as parameters

- the function to be integrated
- the integration variable.

For example

$$\mathcal{L}\ \{e^{-at}\}$$

is entered as

```
laplace_transform(e^(-a*x),x);
```

and returns the result

$$\frac{1}{s+a}$$

5.2 Hankel transform

The Hankel transform

$$f(\omega) = \int_0^\infty F(t)\, J_\nu(2\sqrt{\omega t})\, dt$$

can be calculated by using the `hankel_transform` command e.g.

```
hankel_transform(f(x),x);
```

This is used in the same way as the `laplace_transform` command.

5.3 Y-transform

The Y-transform

$$f(\omega) = \int_0^\infty F(t)\, Y_\nu(2\sqrt{\omega t})\, dt$$

can be calculated by using the `Y_transform` command e.g.

```
Y_transform(f(x),x);
```

This is used in the same way as the `laplace_transform` command.

5.4 K-transform

The K-transform

$$f(\omega) = \int_0^\infty F(t)\, K_\nu(2\sqrt{\omega t})\, dt$$

can be calculated by using the `K_transform` command e.g.

```
K_transform(f(x),x);
```

This is used in the same way as the `laplace_transform` command.

5.5 StruveH transform

The StruveH transform

$$f(\omega) = \int_0^\infty F(t)\, StruveH(\nu, 2\sqrt{\omega t})\, dt$$

can be calculated by using the `struveh_transform` command e.g.

```
struveh_transform(f(x),x);
```

This is used in the same way as the `laplace_transform` command.

5.6 Fourier sine transform

The Fourier sine transform

$$f(s) = \int_0^\infty F(t)\, sin(st)\, dt$$

can be calculated by using the `fourier_sin` command e.g.

```
fourier_sin(f(x),x);
```

This is used in the same way as the `laplace_transform` command.

5.7 Fourier cosine transform

The Fourier cosine transform

$$f(s) = \int_0^\infty F(t)\, cos(st)\, dt$$

can be calculated by using the `fourier_cos` command e.g.

```
fourier_cos(f(x),x);
```

This is used in the same way as the `laplace_transform` command.

6 Additional Meijer G-function Definitions

The relevant Meijer G representation for any function is found by a pattern-matching process which is carried out on a list of Meijer G-function definitions. This list, although extensive, can never hope to be complete and therefore the user may wish to add more definitions. Definitions can be added by adding the following lines:

```
defint_choose(f(~x),~var => f1(n,x);

symbolic putv(mellin!-transforms!*,n,'
              (() (m n p q) (ai) (bj) (C) (var)));
```

where f(x) is the new function, i = 1..p, j=1..q, C = a constant, var = variable, n = an indexing number.

For example when considering $cos(x)$ we have

Meijer G representation -

$$\sqrt{\pi}\, G^{10}_{02}\left(\frac{x^2}{4} \,\middle|\, \begin{matrix} . \;\; . \\ 0 \;\; \frac{1}{2} \end{matrix}\right) dx$$

Internal definite integration package representation -

```
defint_choose(cos(~x),~var)   => f1(3,x);
```

where 3 is the indexing number corresponding to the 3 in the following formula

```
  symbolic putv(mellin!-transforms!*,3,'
                (() (1 0 0 2) () (nil (quotient 1 2))
                (sqrt pi) (quotient (expt x 2) 4)));
```

or the more interesting example of $J_n(x)$:

Meijer G representation -

$$G^{10}_{02}\left(\frac{x^2}{4}\,\middle|\,\begin{matrix} \cdot\ \cdot \\ \frac{n}{2}\ \frac{-n}{2}\end{matrix}\right)dx$$

Internal definite integration package representation -

```
  defint_choose(besselj(~n,~x),~var) => f1(50,x,n);

  symbolic putv(mellin!-transforms!*,50,'
                ((n) (1 0 0 2) () ((quotient n 2)
                                   (minus quotient n 2)) 1
                                   (quotient (expt x 2) 4)));
```

7 The print_conditions function

The required conditions for the validity of the transform integrals can be viewed using the following command:

```
        print_conditions().
```

For example after calculating the following laplace transform

```
        laplace_transform(x^k,x);
```

using the `print_conditions` command would produce

```
repart(sum(ai) - sum(bj)) + 1/2 (q + 1 - p)>(q - p) repart(s)

 and ( - min(repart(bj))<repart(s))<1 - max(repart(ai))

 or mod(arg(eta))=pi*delta

 or ( - min(repart(bj))<repart(s))<1 - max(repart(ai))
```

```
or mod(arg(eta))<pi*delta
```

where

$$
\begin{array}{l}
delta = s + t - \frac{u-v}{2} \\
eta = 1 - \alpha(v-u) - \mu - \rho \\
\mu = \sum_{j=1}^{q} b_j - \sum_{i=1}^{p} a_i + \frac{p-q}{2} + 1 \\
\rho = \sum_{j=1}^{v} dj - \sum_{i=1}^{u} c_i + \frac{u-v}{2} + 1 \\
s, t, u, v, p, q, \alpha \text{ as in (1)}
\end{array}
$$

8 Acknowledgements

I would like to thank Victor Adamchik whose implementation of the definite integration package for REDUCEis vital to this interface.

References

[1] A.P. Prudnikov, Yu.A. Brychkov and O.I. Marichev, *Integrals and Series, Volume 3: More Special Functions* Gordon and Breach Science Publishers (1990)

[2] V.S. Adamchik and O.I. Marichev, *The Algorithm for Calculating Integrals of Hypergeometric Type Functions and its Realization in Reduce System* from *ISSAC 90:Symbolic and Algebraic Computation* Addison-Wesley Publishing Company (1990)

[3] Yudell L. Luke, *The Special Functions and their Approximations, Volume 1* Academic Press (1969).