

Recursive APL Programs for Steffenson's and Romberg's Algorithms

E. Seah
Department of Actuarial and Management Sciences
University of Manitoba

Abstract

We present recursive APL programs for Steffenson's and Romberg's algorithms.

Introduction

In solving for roots of equations and in determining integrals of functions, algorithms that make use of the idea of *extrapolation to the limit* to accelerate the convergence of approximations are often used.

Since APL is now widely used in actuarial work, we present in this note APL programs for two of these algorithms, Steffenson's [1, pp. 61-65] and Romberg's [1, pp. 177-182], which are in the Society of Actuaries Part 3 Examination syllabus. The recursive nature of these algorithms makes it natural to implement them using the recursive construct of APL. Thus, the programs call themselves repeatedly until terminating conditions are reached.

Steffenson's Algorithm

A. Description of the algorithm

This method is used to find roots of equations. An initial (guess) value X_0 is used to generate two values X_1 and X_2 using the iteration formula $X=G(X)$. Hence $X_1=G(X_0)$, and $X_2=G(X_1)$. Based on these three values, Aitken's Δ^2 process is employed to produce a new approximation X_3 , which is then used to produce $X_4 (=G(X_3))$ and $X_5 (=G(X_4))$. This procedure is repeated to generate the sequence $\{X_0, X_3, X_6, X_9, \dots\}$ until two consecutive entries of this sequence differ by less than a predetermined tolerance value.

B. Programs

1. Function STEF with two parameters N and P0.

The parameter N is (an integer giving) the maximum number of iterations allowed before the program is halted for failing to converge to the desired level of accuracy as indicated by the tolerance value. The parameter P0 is the initial approximation (X_0). This program calls STEF1 which does all the calculations.

2. Function STEF1 with two parameters N and P0.

The parameter N is a vector of 2 elements which are both identical to the parameter N in STEF initially. The first element of N stays unchanged throughout, while the second element of N is decreased by one each time an iteration is performed. The parameter P0 is also identical to the parameter P0 in STEF initially. This program calls itself recursively until either the maximum number of allowable iterations is exceeded (in which case the second element of N will become 0) or the sequence converges to the desired level of accuracy.

3. Function G with a parameter X.

This program calculates an iterated value $X_{j+1}=G(X_j)$.

C. Examples

1. Solve for a root of $X^3 + 4X^2 - 10 = 0$, using the iteration formula $G(X)=(10/(X+4))^5$ and $X_0=1.5$.
2. Solve for a root of $X - 2^{-X} = 0$, using the iteration formula $G(X)=2^{-X}$ and $X_0=0$. The tolerance value is set at 0.00001.

D. Refer to Appendix 1 for listings of programs and examples.

Romberg's Algorithm

A. Description of algorithm

It is a method widely used in integration. The trapezoidal rule is used to give preliminary approximations, then Richardson extrapolation is applied to obtain improvements. Referring to the output in the Appendix, we note that the entries of the first column ($R_{11}, R_{21}, R_{31}, \dots$) are the preliminary trapezoidal approximations, where R_{N1} is obtained using 2^{N-1} intervals. The entries in the other columns are generated by the following formula one row at a time:

$$R_{i,j} = \frac{4^{j-1}R_{i,j-1} - R_{i-1,j-1}}{4^{j-1} - 1}$$

B. Programs

1. Function ROMBERG with two parameters A and B.

The parameters A and B specify the range of integration. The program calculates R_{11} (T0) and then calls ROMBERG1 to carry out the rest of the calculations.

2. Function ROMBERG1 with two parameters N and T0.

The parameter N is associated with the number of intervals (2^{N-1}) used in obtaining the trapezoidal approximations. The parameter T0 is the (N+1)th row of the extrapolations from which T1, the (N+2)th row, is to be determined.

ROMBERG1 is terminated when the absolute value $|R_{N+2,N+1} - R_{N+2,N+2}|$ is less than a predetermined tolerance value.

3. Function F with a parameter X.

This program calculates $F(X)$.

C. Examples

1. Integrate $\text{SIN}(X)$ from 0 to π .
 2. Integrate $X(1 + X^2)^5$ from 0 to 3.
- The tolerance value is set at 0.00001.

D. Refer to Appendix 2 for listings of programs and examples.

References

- [1]. Burden, R. L., and J. D. Faires (1985), *Numerical Analysis* (third edition). Prindle, Weber & Schmidt, Boston.

Appendix 1

A STEFFENSON'S ALGORITHM

```

▽STEF[0]▽
▽ N STEF P0
[1] (2/N) STEF1 P0
▽

▽STEF1[0]▽
▽ N STEF1 P0;F1;F2;PX
[1] →(N[2]=0)/FAIL
[2] F1←G P0
[3] F2←G F1
[4] PX←F0-((F1-F0)*2)+F2+F0-2*F1
[5] 5 0 12 7 12 7 12 7 →(-/N),P0,F1,F2
[6] →(TOL);|PX-P0|/END
[7] (N[1],N[2]-1) STEF1 PX
[8] →0
[9] END: 5 0 12 7 →(1+/-/N),PX
[10] →0
[11] FAIL: 'NO. OF ITERATIONS EXCEEDED'
▽
```

TOL:
0.00001

n EXAMPLE 1

```
▽G[ ]▽  
▽ Z+G X  
[1] Z+(10+X+4)*0.5  
▽
```

```
5 STEP 1.5  
0 1.5000000 1.3483997 1.3673764  
1 1.3652652 1.3652255 1.3652306  
2 1.3652300 1.3652300 1.3652300  
3 1.3652300
```

n EXAMPLE 2

```
▽G[ ]▽  
▽ Z+G X  
[1] Z+2*-X  
▽
```

```
10 STEP 0  
0 .0000000 1.0000000 .5000000  
1 .6666667 .6299605 .6461941  
2 .6412162 .6411722 .6411918  
3 .6411857 .6411857 .6411857  
4 .6411857
```

Appendix 2

n ROMBERG'S ALGORITHM

```
▽ROMBERG[ ]▽  
▽ A ROMBERG B;T0  
[1] 12 7 +T0+ ,+/(B-A)*0.5xF A,B  
[2] 0 ROMBERG1 T0  
▽
```

```

▽ROMBERG1[0]▽
▽ N ROMBERG1 T0;T1;H;I;5
[1] T1+0.5xT0[I]+Hx+/F A+(-0.5+(2xN)xH+(B-A)÷2xN
[2] I+1
[3] LP;+((1+T0)(I+I+1)/CHK
[4] T1+T1,((5x-1↑T1)-T0[I-1])÷-1+5+4xI-1
[5] +LP
[6] CHK; 12 7 ↑T1
[7] +(TOL)|(↑T1)-↑T0)/0
[8] (N+1) ROMBERG1 T1
▽

```

A EXAMPLE 1

```

▽F[0]▽
▽ Z+F X
[1] Z+10X
▽

```

```

0 ROMBERG 01
.0000000
1.5707963 2.0943951
1.8961189 2.0045598 1.9985707
1.9742316 2.0002692 1.9999831 2.0000055
1.9935703 2.0000166 1.9999998 2.0000000 2.0000000

```

A EXAMPLE 2

```

▽F[0]▽
▽ Z+F X
[1] Z+Xx(1+Xx2)*0.5
▽

```

```

0 ROMBERG 3
14.2302495
11.1713699 10.1517434
10.4437968 10.2012725 10.2045744
10.2663672 10.2072240 10.2076207 10.2076691
10.2222702 10.2075712 10.2075943 10.2075939 10.2075936
10.2112607 10.2075909 10.2075922 10.2075922 10.2075922 10.2075922

```