

Numerical Implementation of the Natural Exponential

Alberte Mørk

There are several ways of defining the natural exponential, $\exp(x)$. One could define it as the solution of the differential equation $y' = y$ or as the limit $\lim_{n \rightarrow \infty} (1 + x/n)^n$. They all turn out to be equal. The most common way, and the way most relevant for our purpose, is by the power series,

$$\exp(x) := \sum_{k=0}^{\infty} \frac{x^k}{k!}.$$

With a bit of knowledge of complex function theory it is rather straightforward to show that the convergence radius of this function is infinite, making the function well-defined on all of \mathbb{C} .

The definition is handy in the framework of numerics is because it only consists of basic operations addition and multiplication. To exploit this feature, we rewrite the sum as

$$\exp(x) = 1 + x \left(1 + \frac{x}{2} \left(1 + \frac{x}{3} (\dots) \right) \right).$$

By making this rewriting we avoid power functions, which are slower than ordinary multiplication. Another perk is that the computer will start the calculation from the innermost bracket, such that the smallest contributions are calculated first. This becomes a problem for negative values, as these will have the largest contribution first. To overcome this issue, we define the function recursively to return the inverse of the function taken on the positive value if the input is negative.

```
if  $x < 0$  then return  $\frac{1}{\exp(-x)}$   
end if
```

The next issue that arises, is the fact that we cannot start at infinity. This we address by noting that for small input values, the higher order terms are negligibly small. To ensure that the argument is small we implement another recursive algorithm.

```
if  $x < \epsilon$  then return  $\exp\left(\frac{x}{2}\right)^2$   
end if
```

The value ϵ is chosen to be small. For higher precision, lower values of ϵ is used. This finally allows us to define the function

```

function exp( $x$ )
  if  $x < 0$  then return  $\frac{1}{\exp(-x)}$ 
  end if
  if  $x < \epsilon$  then return  $\exp\left(\frac{x}{2}\right)^2$ 
  end if return  $1 + x\left(1 + \frac{x}{2}\right)$  (to order  $n$ )
end function

```

We have made a specific implementation using $\epsilon = 1/8$ and $n = 10$. This gives us the plot below, which shows a good correspondence between the numerical routines, and the routine from math.h.

