1 - INTRODUCCION

El análisis numérico básicamente se encarga de analizar, describir y crear algoritmos numéricos que permiten resolver problemas matemáticos. Estos algoritmos generalmente nos permiten obtener resultados aproximados, ya que contienen un número finito de pasos. El uso del análisis numérico toma gran importancia con el uso de las computadoras y el poder de calculo que ellas tienen. Por este medio, es posible resolver problemas mas complejos.
Pero el uso de computadoras para hacer cálculos complejos trae un problema con sigo, y surge el concepto de error. Este concepto nace debido a que las computadoras trabajan con un rango finito de números, y además cada uno de estos está representado de una forma también finita.

Los errores están divididos en trés tipos: Errores en los datos de entrada, errores de redondeo y errores de truncamiento. Los errores en los datos de entrada no están causados por el algoritmo que resuelve el problema, sino por valores que inician el algoritmo, generalmente estos valores se refieren a mediciones o magnitudes físicas. Los errores de redondeo surgen cuando se utilizan operaciones que tienen una representación numerica finita, esto significa que tienen una precisión limitada con respecto al resultado que devuelven. Y los errores de truncamiento están relacionados con el algoritmo en si, esto quiere decir que dependen de la forma en que se resuelve el problema, en algunos casos el error de truncamiento se puede disminuir modificando o refinando el algoritmo, esto generalmente implica aumentar la cantidad de operaciones a hacer y por lo tanto aumentar el error de redondeo y el tiempo para resolver el problema.

El uso de computadoras en el cálculo numérico no solo conduce a errores de los antes mencionados, sino que, impulsado por alguno de los anteriores, aparece el error por resolver el probléma no como se ha formulado, sino a través de alguna aproximación. Este error es causado por reemplazar un infinito(sumatoria o integral por ejemplo) por una cantidad finita de términos

La precisión finita que introduce algunos de los errores numéricos, es la que opera bajo el estándar de la IEEE (Institute of Electrical and Electronics Engineers) nro. 754, que normaliza la notación en bits de los numeros de punto flotante, la norma tiene cuatro grados de precisión: simple (32 bits), simple extendida(43 bits, no se utiliza habitualmente), doble (64) y doble extendida (implementada en 80 bits o más). Todas las normas cuentan con tres campos: signo, mantisa y exponente, los dos últimos varian su longitud según de cual se trate. En el caso de la doble extendida, se asigna un bit al signo (la parte más alta de la cadena), 64 a la mantisa (que se almacena en la parte baja de la cadena en notación sin signo) y 15 al exponente (entre las otras dos, se encuentra desplazado 2^(14)). El rango representado es [-10^(4932), 10^(4932)] aprox.

A su vez, existen varias maneras de cuantificar el error, de forma que su medición se torne algo más tangible y útil, entre ellas, el error relativo y el error absoluto son las que trataremos en este trabajo. Desde el punto de vista absoluto, el error surge de la diferencia que pudiera haber entre la magnitud real que se desea expresar, y la obtenida mediante cálculos computacionales. Por otra parte el error relativo tiene en cuenta no solo las magnitudes, sino el cociente entre la diferencia anterior y el valor absoluto de la magnitud medida. De esta forma se obtiene una real dimensión del error y no tan solo el valor absoluto del mismo.

Por estos motivos, una de las tareas del análisis numerico se trata de buscar un algoritmo que lleve a la mejor solución posible de cada problema. De esta forma se pueden definir muchos algoritmos que lleguen a la solución, pero se seleccionará el que mejor aproxime a la solución del problema, es decir, el que mejor utilice las operaciones respecto de sus errores.

En este caso surge la noción de estabilidad numérica. La estabilidad numérica define cuan buena será la solución de nuestro problema usando métodos aproximados. Estos métodos pueden tener un resultado diferente al esperado, ya que tienen diferente estabilidad numerica, esto quiere decir que para ciertos valores de entrada, con sus respectivos errores, el método puede propagar el error por el algoritmo en mayor o menor medida. De esta forma, el algoritmo que mejor aproxime a la solución del problema, será aquel que tenga mejor estabilidad numerica.

2 - DESARROLLO

El problema puntual abordado por este trabajo es obtener una aproximación de la función e^(-x), mediante el polinomio de Taylor. El desarrollo de este algoritmo contempla dos variantes

[INSERTAR PARTE DEL ENUNCIADO DEL TP aquí]

En cualquier caso, se incurre en varios tipos de error, a saber:

1) error debido a la implementación en aritmetica finita
2) error de truncamiento en cuanto a que el polinomio de Taylor es una serie, por tanto infinita, de términos, que se verá acotada a la cantidad que el usuario considere conveniente.

Es condición para el trabajo realizar la aritmetica con precisión arbitraria (no por eso pierde su calidad de finita), determinada por el usuario. Entiéndase por precisión, el numero de bits de la mantisa en notación normalizada, esta cantidad esta acotada (en este trabajo) superiormente por 64.
El resultado de analizar las magnitudes y los comportamientos de los errores introducidos por el polinomio de Taylor para el calculo de la función serán abordados en este mismo trabajo, más adelante.

2.1- DETALLES DE IMPLEMENTACION