Universidad Nacional Autónoma de Nicaragua, Managua

Recinto Universitario Rubén Darío

Facultad de Educación e Idiomas

Departamento de Matemática

Tesis Monográfica para optar al Grado de

Doctor en Matemática Aplicada

”Algoritmos basados en los Polinomios de Adomian e Iteración Variacional para la resolución de ecuaciones no lineales"

Autor

Msc. Iván Augusto Cisneros Díaz

Tutor: Dr. Ramón Antonio Parajón Guevara

Managua, Agosto 2017

Dr. Ramón Antonio Parajón Guevara, docente titular del Departamento de Matemática de la Facultad de Educación e Idiomas, de la Universidad Nacional Autonoma de Nicaragua, UNAN-Managua.

CERTIFICA que la presente Monografía

"Algoritmos basados en los Polinomios de Adomian e Iteración Variacional para la resolución de ecuaciones no lineales"

ha sido realizada bajo su dirección en el Departamento de Matemática por el Magister

Iván Augusto Cisneros Díaz

y constituye su tesis para optar al grado de Doctor en Matemática Aplicada.

Y para que así conste, en cumplimiento con la normativa vigente, autoriza su presentación ante la Coordinación del Programa de Doctorado, para que pueda ser tramitada su lectura y defensa pública.

Tutor de Tesis

Dr. Ramón Antonio Parajón Guevara

Managua, Nicaragua 2017.

Dedicatoria

A los niños que despiertan buscando el amanecer de la sabiduría de las Ciencias.

A los jóvenes que predican el conocimiento a través de la búsqueda de la práctica científica.

A todas las personas que día a día, construyen el conocimiento, obteniendo la felicidad de ser autodidacta.

Agradecimiento

A los Maestros, por su infinita paciencia, en la transmisión de los conocimientos.

A las Ciencias por permitir descubrir diferentes aspectos de la realidad y verdades ocultas.

Índice

1. RESUMEN **1**

2. INTRODUCCIÓN **2**

3. PLANTEAMIENTO DEL PROBLEMA **5**

4. METODOLOGÍA DESARROLLADA **6**

5. OBJETIVOS **7**

5.1. Objetivo General 7

5.2. Objetivos Específicos 7

6. CAPÍTULO I : CONCEPTOS GENERALES **8**

6.1. Introducción 8

6.2. Estado del Arte 9

6.3. Conceptos Básicos 11

6.4. Método del Punto Fijo 21

6.5. Métodos Iterativos 31

7. CAPÍTULO II : MÉTODO DE NEWTON **36**

7.1. Análisis de Convergencia del Método de Newton 36

7.2. Variantes del Método de Newton 37

7.2.1. Caso I : Forma funcional $h(x) = f(x) + g(x)$ 37

7.2.2. Caso II : Forma funcional $h(x) = f(x) - g(x)$ 40

7.2.3. Caso III : Forma funcional $h(x) = f(x)\, g(x)$ 42

7.2.4. Caso IV : Forma funcional $h(x) = f(x)/g(x)$ 45

7.3. Resultados Numéricos 47

7.4. Conclusiones 50

8. CAPÍTULO III : POLINOMIOS DE ADOMIAN **52**

8.1. Construcción del Método de Newton por el método de Adomian 53

8.2. Método iterativo ABBS 54

8.3. Método iterativo de Chun 55

8.4. Variantes de Métodos Iterativos 57

8.5. Resultados Numéricos . . . 77
8.6. Conclusiones . . . 80

9. CAPÍTULO IV : ITERACIÓN VARIACIONAL **81**
9.1. Caso I : $H(x) = x + \lambda f(x) g(x)$ . . . 81
9.2. Caso II. $H_1(x) = \phi(x) + \lambda (f(x) g(x))^p$, $p = 1$ . . . 84
9.3. Caso III : $H(x) = \phi(x) - \dfrac{\phi'(x)(f(x) g(x))}{(f(x) g(x))'}$ . . . 88
9.4. Caso IV : $H(x) = \phi(x) - \dfrac{\phi'(x)(f(x) g(x))}{p(f'(x) g(x) + g'(x) f(x))}$ . . . 92
9.5. Caso V : $H(x) = \phi(x) + \lambda (f(x) g(x))^p$, $p = 2$ . . . 102
9.6. Caso VI : $H(x) = \phi(x) + \lambda (f(\phi(x)) g(\phi(x)))$ . . . 114
9.7. Caso VII : $H(x) = \phi(x) + \lambda (f(\psi(x)) g(\psi(x)))^p$ . . . 130
9.8. Resultados Numéricos . . . 131
9.9. Conclusiones . . . 135

10. CAPÍTULO V : ANÁLISIS DE CONVERGENCIA **137**
10.1. Convergencia del esquema $H(x) = \phi(x) - \dfrac{\phi'(x)(f(x) g(x))}{p(f'(x) g(x) + g'(x) f(x))}$ . . . 137
10.2. Convergencia del esquema $x_n = y - \dfrac{f(y) g(x)}{f'(x) g(x) + g'(x) f(x)}$ . . . 139
10.3. Convergencia del esquema $x_n = z - \dfrac{f(z) g(x)}{f'(x) g(x) + g'(x) f(x)}$ . . . 141
10.4. Convergencia del esquema $x_n = y - \dfrac{f(x) f(y) g(x)}{f'(y)[f(x) - 2f(y)] + f(x) f(y) g'(x)}$ . . . 143

11. CONCLUSIONES GENERALES **146**

12. REFERENCIAS BIBLIOGRÁFICAS **148**

13. ANEXO **156**
13.1. Método de Newton . . . 156
13.2. Polinomios de Adomian . . . 157
13.3. Técnica Iterativa Variacional . . . 158
13.4. Esquemas Iterativos Complementarios . . . 159
13.5. Código Fuente del Algoritmo Newton . . . 162
13.6. Gráficas Funcionales . . . 164
13.7. Raíz obtenida por los algoritmos . . . 165

1. RESUMEN

Esta tesis aborda las técnicas de los polinomios de Adomian e Iteración Variacional, que son métodos iterativos para resolver ecuaciones no lineales de la forma $f(x) = 0$. El objetivo principal es generar nuevos algoritmos y nuevos esquemas iterativos que permitan obtener nuevas fórmulas y métodos iterativos. Se estudian los polinomios de Adomian y se construyen nuevas variantes del método de Newton. También se estudian la técnica iterativa variacional y se obtienen algunos resultados conocidos, como también, nuevos esquemas y por ende, nuevos métodos iterativos.

En el presente estudio se realiza una revisión de las diversas fórmulas existentes y se crean nuevas fórmulas mediante procedimientos matemáticos basados en los polinomios de Adomian y la técnica iterativa variacional.

Se desarrolla la construcción de los principales esquemas iterativos, asi como el análisis de su convergencia, enfatizando en el orden de convergencia de dicho método.

Este estudio permitió obtener los principales esquemas iterativos de cada método, mediante la deducción de su método constructivo, asi como el análisis de convergencia del mismo. Se ejemplifican y se calculan raíces de funciones no lineales de algunas funciones bases, utilizadas en los artículos científicos consultado.

También, se realiza una comparación entre los algoritmos existentes y los diseñado en nuestra investigación, utilizando los criterios de: orden de convergencia, eficiencia computacional, índice operacional, así como el máximo y mínimo número de evaluaciones funcionales e índice de eficiencia computacional.

Según los resultados obtenidos después de las comparaciones, nuestros algoritmos presentan un excelente funcionamiento con respecto a los existentes en la literatura sobre este área de conocimiento.

2. INTRODUCCIÓN

En la vida real existen existen diversas clases de problemas que surgen en la teoría de la optimización, control económico, robótica, inteligencia artificial, teoría de ecuaciones, modelos de la física-matemática y en otras áreas de conocimiento que se puede formular utilizando una ecuación no lineal del tipo

$$f(x) = 0$$

Esta ecuación puede ser una ecuación algebraica de orden superior y posiblemente puede implicar términos trigonométricos, trigonométricos inversos, exponenciales, logarítmicos, hiperbólicos, hiperbólicos inversos o ser completamente una ecuación trascendente.

Se sabe que toda ecuación algebraica de orden mayor o superior a cinco, no puede ser resuelta bajo ningún método de radicales y en algunos casos, no se puede encontrar la solución exacta de dicha ecuación. Para superar esta dificultad, se han desarrollado métodos numéricos que permiten obtener soluciones aproximadas de dicha ecuación no lineal.

La resolución de ecuaciones y sistemas de ecuaciones no lineales figura entre los problemas más importantes, tanto desde un punto de vista teórico como práctico, de las matemáticas aplicadas, así como también de muchas ramas de las ciencias, la ingeniería, la física, la informática, la astronomía, las finanzas, etc. Un vistazo a la bibliografía y la lista de grandes matemáticos que han trabajado en este tema pone de manifiesto un alto nivel de interés contemporáneo en el mismo.

Aunque el rápido desarrollo de las computadoras digitales llevó a la aplicación efectiva de muchos métodos numéricos, en la realización práctica, es necesario analizar diferentes problemas tales como: la eficiencia computacional basado en el tiempo usado por el procesador, el diseño de métodos iterativos que posean una rápida convergencia a la solución deseada, el control de errores de redondeo, la información sobre las cotas de error de la solución aproximada obtenida, las condiciones iniciales que garanticen una convergencia segura, número de evaluaciones funcionales, la eficiencia computacional, índice computacional, índice operacional, orden de convergencia, número de iteraciones, etc. Dichos problemas constituyen el punto de partida de este trabajo.

En la actualidad, el estudio de los métodos iterativos para resolver ecuaciones no lineales, es un campo bastante explorado por la investigación de matemática numérica y cada dia, la preocupación es más arraigada, debido a la imperante necesidad de resolver ecuaciones no lineales que no poseen soluciones analíticas exactas.

La inquietud por mejorar el orden y la rapidez de convergencia de los métodos iterativos, es una necesidad imperante en el estudio de dichos métodos y cada dia, es más creciente las investigaciones referidas a la búsqueda de nuevos procesos numéricos.

Entre los principales valores metodológicos de la tesis, se pueden mencionar :

1. Diseñar nuevos algoritmos para obtener soluciones de ecuaciones no lineales.

2. Aplicación de diferentes técnicas numéricas creadas para la solución de una ecuación no lineal.

3. Evidenciar la importancia de la ciencia computacional en la búsqueda de nuevos enfoques de paradigmas de programación matemática.

Esto reafirma, el objetivo central de este trabajo, el cual consiste en diseñar nuevos algoritmos, basados en la Descomposición de los Polinomios de Adomian y la Técnica Iterativa Variacional, los cuales permiten obtener nuevas variantes del método de Newton.

Los objetivos especificos alcanzado al desarrollar esta tesis son:

1. Diseñar nuevos métodos iterativos para resolver ecuaciones no lineales.

2. Estudiar la convergencia de los nuevos métodos iterativos desarrollados.

3. Implementar en lenguaje C++, los nuevos métodos numéricos estudiados.

El capítulo 1, se refiere a los conceptos generales de la teoría de los métodos iterativos, enfatizando en aquellas definiciones y teoremas generales necesarios para la investigación, también presenta el estado de arte de la investigación, con el objetivo de mostrar la amplitud de las diversas técnicas numéricas y poder dirigir nuestro estudio a la continuación de los conocimientos que se han alcanzado. También se presenta un listado de definiciones, propiedades y teoremas relativos a los métodos iterativos, destacando el método general del punto fijo y sus principales consecuencias teórica y a partir de aqui, se infiere el caso particular del método de Newton. También se enfatiza en los aspectos relativos al análisis y orden de convergencia.

El capítulo 2, muestra una perspectiva del método de Newton, resaltando nuevos esquemas iterativos y obteniendo nuevas fórmulas iterativas para el hallazgo de las raíces reales de ecuaciones no lineales. Se analiza su orden de convergencia y se presenta diversas variantes del mismo.

El capítulo 3, destaca la teoría de los polinomios de Adomian, sus diversas variantes, asi como, los métodos iterativos de Abbasbandy y el método de Chun. Se desarrolla la construcción de ambos método iterativo, asi como, el análisis de su convergencia. En este acápite, se obtienen nuevos esquemas y métodos iterativos basado en la técnica de los polinomios de Adomian.

El capítulo 4, desarrolla la teoría de la Técnica Iterativa Variacional, sus diversas variantes, asi como la obtención de nuevos esquemas y métodos iterativos.

El capítulo 5, se demuestran los teoremas generales de los diversos esquemas iterativos obtenidos por la técnica de variacion iterativa y se presentan algunos casos especiales de los mismos. Se estudia el orden de convergencia de los diferentes esquemas iterativos.

Los capítulos 2 - 4, desarrollan los resultados numéricos obtenidos con los diversos algoritmos, destacando aquellos que superan o son equivalente al método de Newton, de acuerdo al número de iteraciones necesarias para alcanzar dicha raíz real. Es necesario señalar que la selección de estas funciones bases, obedecen a criterios de comparación que se efectuaron entre las aproximaciones numéricas presentadas en los artículos científicos consultados y los obtenidos por los diversos programas que se codificaron para el análisis de la información. También estos capítulos contienen una conclusión del resultados de las ejecuciones de los códigos fuentes desarrollados para cada uno de dichos algoritmos.

La programación de todos los algoritmos fueron codificados en el lenguaje de alto nivel C++, para este fin, se utilizó técnicas de POO (Programación Orientada a Objetos), es decir, programación que emplea definición de clase.

Varias técnicas numéricas se han introducido para desarrollar nuevos métodos numéricos, entre ellos: Serie de Taylor [8, 57, 94], fórmulas basadas en la cuadratura de Gauss [63], Perturbación homotópica [67, 68], Descomposición de Adomian [1, 2, 10, 42] y la técnica de iteración variacional [34, 65], todos ellos han sido desarrollados e implementados para resolver ecuaciones no lineales.

3. PLANTEAMIENTO DEL PROBLEMA

La resolución de ecuaciones y sistemas de ecuaciones no lineales figura entre los problemas más importantes, de las matemáticas aplicadas, la ingeniería, la física, la informática, la astronomía, las finanzas, etc. Un vistazo a la bibliografía y la lista de grandes matemáticos que han trabajado en este tema pone de manifiesto un alto nivel de interés contemporáneo en el mismo.

El estudio de los métodos iterativos para resolver ecuaciones no lineales es cada día mas importante, ya que es necesario obtener soluciones aproximadas confiables, necesarias para resolver problemas de la vida real.

La necesidad por mejorar el orden y la rapidez de convergencia de los métodos iterativos es muy imperante en el estudio de dichos métodos y cada día, es mas creciente las investigaciones referidas a la búsqueda de nuevos procedimientos numéricos.

Por lo antes expuesto, nuestro interés se centra en mejorar y optimizar las soluciones aproximadas para el problema de la resolución de ecuaciones no lineales, mediante la descomposición de los polinomios de Adomian y la técnica iterativa variacional.

4. METODOLOGÍA DESARROLLADA

La metodología empleada en este trabajo siguió las siguientes etapas:

1. Se revisó el estado del arte sobre los métodos iterativos existentes, la cual consistió en analizar la metodología y resultados obtenidos en 307 artículos publicados en revistas indexadas de matemáticas aplicadas. Esto permitió obtener una visión general sobre las diferentes variantes existentes sobre el método de Newton, además de apreciar la belleza teórica con que es abordado este tema de matemática numérica.

2. Se estudiaron los procedimientos matemáticos de los Polinomios de Adomian y de la Técnica Iterativa Variacional, asi como sus diferentes aplicaciones a la teoría de los métodos iterativos.

3. Generación de diversos esquemas iterativos y búsqueda de funciones arbitrarias auxiliares que permitieran obtener nuevos métodos iterativos, basados en los polinomios de Adomian y la Técnica Iterativa Variacional.

4. Se formularon y demostraron teoremas que permiten obtener el orden de convergencia de dichos métodos numéricos.

5. Programación en C++, de todos los algoritmos consultados y los nuevos obtenidos por las técnicas de los polinomios de Adomian y la variación iterativa.

6. La comparación realizada se basó en el número de iteraciones de cada método para determinar la solución real de la ecuación no lineal.

5. OBJETIVOS

5.1. Objetivo General

Diseñar nuevos algoritmos basado en la descomposición de los polinomios de Adomian y en la técnica de iteración variacional.

5.2. Objetivos Específicos

1. Generalizar el método de Newton, basado en diferentes enfoque matemáticos y métodos aproximativos para la solución de ecuaciones no lineales.

2. Aplicar la teoría de los polinomios de Adomian y la técnica de variación iterativa para generar nuevos esquemas y métodos iterativos.

3. Diseñar los algoritmos y programas computacionales de los nuevos métodos iterativos obtenidos, evaluándolo con algunos criterios comparativos y clasificativos.

6. CAPÍTULO I : CONCEPTOS GENERALES

6.1. Introducción

El objetivo principal de este capítulo es introducir algunos conceptos básicos referentes a las principales definiciones, propiedades y teoremas relativos a la teoría de métodos iterativos, tales como: orden de convergencia, multiplicidad de una raíz y eficiencia computacional. También se presentan algunos resultados de la teoría general de los métodos iterativos necesarios para el estudio a realizarse en los capítulos posteriores.

En la actualidad existen una gran cantidad de métodos iterativos para resolver ecuaciones no lineales, con $f : \mathbb{R} \rightarrow \mathbb{R}$, una función real de variable real. La esencia de estos métodos consiste en que si se conoce un entorno suficientemente pequeño que contiene solo a una raíz α de la ecuación $f(x) = 0$ y seleccionamos una estimación inicial de la raíz x_0, lo suficientemente cerca de la raíz buscada α, generamos mediante una función de punto fijo $g(x)$ una sucesión de iterados $x_1,\ x_2,\ \cdots\ ,\ x_k\ ,\ \cdots$ que convergen a la raíz α. La convergencia de la sucesión se garantiza con la elección apropiada de la función g y de la aproximación inicial x_0.

El método descrito por la función de iteración $g : I \subseteq \mathbb{R} \rightarrow \mathbb{R}$ tal que

$$x_{k+1} = g(x_k) \quad , \quad k = 0, 1, 2, \ \cdots$$

partiendo de una estimación inicial x_0 dada, incluye una gran cantidad de esquemas iterativos, los cuales difieren entre si por la forma de definir la función de iteración g. Para el estudio de la convergencia de los métodos iterativos, asi como para probar la existencia de la raíz de la ecuación $f(x) = 0$ se usan las llamadas funciones contractivas.

Un método iterativo es un método que progresivamente va calculando aproximaciones a la solución de un problema. En Matemática, un método iterativo se repite un mismo proceso de mejora sobre una solución aproximada: se espera que lo obtenido sea una solución más aproximada que la inicial. El proceso se repite sobre esta nueva solución hasta que el resultado más reciente satisfaga ciertos requisitos. A diferencia de los métodos directos, en los cuales se debe terminar el proceso para tener la respuesta, en los métodos iterativos se puede suspender el proceso al término de una iteración y obtener una aproximación a la solución.

Un elemento en contra que tienen los métodos iterativos sobre los métodos directos es que calculan aproximaciones a la solución. Los métodos iterativos se usan cuando no se conoce un método para obtener la solución en forma exacta. También se utilizan cuando el método para determinar la solución exacta requiere mucho tiempo de cálculo, cuando una respuesta aproximada es adecuada, y cuando el número de iteraciones es relativamente reducido.

Un método iterativo consta de los siguientes pasos:

1. Inicia con una solución aproximada (semilla).
2. Ejecuta una serie de cálculos para construir una mejor aproximación partiendo del valor de la semilla. La fórmula iterativa que permite construir la nueva aproximación usando la anterior, se conoce como ecuación de recurrencia.
3. Se repite el paso anterior pero usando como semilla la nueva aproximación obtenida.

6.2. Estado del Arte

Los métodos iterativos se han desarrollado en los últimos siglos debido a su eficacia para afrontar diferentes problemas no lineales, tales son los casos de predicciones de órbitas planetarias, hasta los procesos más recientes utilizado en la inteligencia artificial, específicamente en el campo de la robótica.

La clasificación de los esquemas iterativos para resolver ecuaciones no lineales del tipo $f(x) = 0$ se basa en el número de pasos del proceso: así, un método puede ser de un solo paso o multipaso. El más conocido método de un solo paso es el de Newton, cuya expresión iterativa es

$$x_{k+1} = x_k - \frac{f(x_k)}{f'(x_k)}$$

en cada paso del proceso, este método necesita dos evaluaciones funcionales $(f,\ f')$. Bajo ciertas condiciones, el orden de convergencia del método de Newton es dos, lo que proporciona una buena eficiencia con un coste computacional razonable, de ahí su amplio uso. Sin embargo, sucesivos intentos de mejorar su eficiencia, en términos de velocidad de convergencia, dieron sus frutos en forma de otros métodos de un solo punto: los esquemas de Halley, Chebyshev, etc. Los cuales tienen convergencia cúbica, donde sus expresiones iterativas tienen un elemento en común: el operador de convexidad logarítmica

$$L_f(x_k) = \frac{f(x_k)\, f''(x_k)}{[f'(x_k)]^2}$$

el uso de este operador permite acelerar la convergencia hasta tercer orden, pero a costa de evaluar la segunda derivada de la función no lineal en cada iterado. Esta idea puede extenderse para acelerar aun más la velocidad de convergencia, pero a costa de evaluaciones funcionales de derivadas de órdenes superiores, lo que a efectos prácticos no es conveniente.

Esta y otras limitaciones de los métodos punto a punto llevan, en la segunda mitad del siglo XX, al desarrollo de métodos multipaso que, mediante expresiones iterativas sencillas y evaluaciones de derivadas de bajo orden de la función no lineal permiten obtener órdenes de convergencia elevados. Un esquema propuesto por Traub, es el método de Potra-Ptak, el cual su expresión iterativa parte del método de Newton como primer paso y un segundo paso consiste en aplicar Newton sobre el primero, pero evaluando únicamente la función en el primer paso y manteniendo la derivada evaluada en el iterado anterior, es decir

$$\begin{aligned} y_k &= x_k - \frac{f(x_k)}{f'(x_k)} \\ x_{k+1} &= y_k - \frac{f(y_k)}{f'(y_k)} \end{aligned}$$

este método tiene orden de convergencia tres, posteriormente se desarrollaron los métodos clásicos de la familia Chebyshev-Halley. Jarratt, Ostrowski y los métodos de King, los cuales no hacían uso de la evaluación de la derivada segunda, todo ellos con orden de convergencia cuatro.

Durante las siguientes décadas se profundizó en el conocimiento de estos métodos, analizando su convergencia local mediante la determinación de regiones donde se aseguraba su convergencia. Sin embargo, el trabajo de Weerakoon y Fernando [21] basado en un trabajo previo de Ozban, marca un nuevo punto de inicio en esta área de investigación. En este momento comienza el auge en el diseño de métodos iterativos multipasos para la resolución de ecuaciones no lineales. En la última década del siglo XX y la primera del siglo XXI aparecen numerosos trabajos en los que se diseñan tantos métodos como familias de métodos óptimos de órdenes crecientes. Para ellos se utilizan diferentes técnicas, pero con una raíz común: la composición de métodos conocidos (básicamente Newton) y la posterior eliminación de algunas de las evaluaciones funcionales añadida en el proceso. La razón del éxito de esta técnica es simple: la composición de dos métodos de órdenes p y q respectivamente, permite construir un nuevo esquema cuyo orden de convergencia es el producto de los órdenes de los métodos originales. Es por tanto, una técnica que permite aumentar la velocidad de convergencia, sin embargo, aumenta el número de evaluaciones funcionales implicadas. Para evitar esto, se elimina alguna de las evaluaciones funcionales añadida en el proceso de composición, ya sea por interpolación polinómica, por funciones racionales o por polinomios de Adomian.

Otra técnica reciente que permite aumentar significativamente el orden de convergencia, sin elevar el número de evaluaciones funcionales, es la técnica de las funciones peso, en el proceso de composición, se sustituye en el último paso una o más de las evaluaciones funcionales nuevas por otras existentes, con lo que el método resultante no alcanza el producto de los órdenes de partida. Para hacerlo, se introduce como factor una función indeterminada cuya variable contiene alguna de las evaluaciones funcionales usadas en el método.

Finalmente se analizan métodos iterativos multipunto óptimos con derivada para la solución de ecuaciones no lineales, donde dichos esquemas son construidos mediante la técnicas del polinomio de Adomian, Técnicas de Variación Iterativas y esquema predictor-corrector, donde estos últimos tienen la estructura del esquema de Potra-Ptak.

6.3. Conceptos Básicos

Definición 1 *Dada una sucesión infinita de elementos* $\{x_i\}_{i=1}^{\infty}$*del espacio métrico* (E, d) *se dice que la sucesión es convergente hacia el elemento* $x^* \in E$*, si para cualquier valor* $\varepsilon > 0$ *siempre se puede encontrar un número natural* N *tal que para todo* $n > N$ *se verifica que* $d(x_n, x^*) < \varepsilon$*. Al elemento* x^* *se le denomina, si existe, límite de la sucesión* $\{x_i\}_{i=1}^{\infty}$*. En forma simbólica*

$$\lim_{n\to\infty} x_n = x \Longleftrightarrow \forall\ \varepsilon > 0,\ \exists\ N \in \mathbb{N}\ tal\ que\ \ n \geq N\ implica\ que\ \ \ |x_n - x| < \varepsilon$$

La notación $\lim_{n\to\infty} x_n = x$ significa que la sucesión $\{x_n\}_{n=1}^{\infty}$ converge a x

Definición 2 *Dada una sucesión infinita de elementos* $\{x_i\}_{i=1}^{\infty}$ *del espacio métrico* (E, d) *se dice que la sucesión es una sucesión de Cauchy, si para cualquier valor* $\varepsilon > 0$ *siempre se puede encontrar un número natural* N *tal que* $\forall\ n > N$ *y* $m > N$ *se verifica que* $d(x_n, x_m) < \varepsilon$*.*

Definición 3 *El espacio métrico* (E, d) *es un espacio métrico completo si toda sucesión de Cauchy de elementos de* E *es una sucesión convergente en* (E, d)*.*

Definición 4 *Sea* g *una aplicación definida en un espacio métrico* (E, d) *y con valores en el mismo espacio métrico, se denomina punto fijo de la aplicación* g *a todo elemento* x^* *de* E *tal que* $x^* = g(x^*)$*. Geométricamente un punto fijo representa el punto de intersección de la curva* $y = g(x)$ *con la recta* $y = x$*.*

Definición 5 *Sean (E,d) y (V,d^*) dos espacios métricos y sea $g: E \to V$ una aplicación definida en E y con valores en V. Se dice que g es una aplicación lipschitziana de razón k cuando existe una constante real $k > 0$ tal que:*

$$d^*(g(x), g(y)) \leq k\ d(x,y), \quad \forall\ x, y \in E$$

A la menor constante k que verifica la condición anterior se la denomina constante de Lipschitz (o razón) de la aplicación. Hay un tipo de funciones lipschitzianas de interés especial, las funciones contractivas, que son funciones lipschitzianas para las cuales k toma valores $0 < k < 1$.

Teorema 6.1 *Toda aplicación lipschitciana definida en (E,d) y con valores en (V,d^*) es una aplicación continua en todo E.*

Demostración 1 *Si $g: E \to V$ es una aplicación lipschitciana con constante de Lipschitz k se verifica que*

$$d^*(g(x), g(y)) \leq k\ d(x,y), \quad \forall\ x, y \in E$$

Por tanto para cualquier valor de ε estrictamente positivo y para cualquier punto x de E se tiene que

$$\forall\ \varepsilon > 0\ ,\ \exists\ \delta = \frac{\varepsilon}{k}\ ,\ d(x,y) < \delta \Longrightarrow d^*(g(x), g(y)) \leq k\ d(x,y) < \varepsilon$$

Por tanto g es continua en todo punto $x \in E$.

Definición 6 *A toda aplicación lipschitciana que verifique las dos condiciones siguientes:*

1. Estar definida en un espacio métrico (E,d) sobre si mismo: $g: E \to E$

2. Tener una constante de Lipschitz estrictamente inferior a 1.

se le denomina contracción sobre E.

Teorema 6.2 *Toda contracción definida sobre un espacio métrico completo admite un único punto fijo.*

Demostración 2 *Comencemos demostrando la existencia del punto fijo. Sea* $g : E \to E$ *una contracción, de constante de Lipschitz* $k < 1$*, definida en el espacio métrico* (E, d) *y sea* x_0 *un elemento cualquiera de* E*. Considérese entonces la sucesión formada a partir de* x_0 *mediante* $x_{i+1} = g(x_i)$

Para la sucesión $\{x_i\}_{i=1}^{\infty}$ se verifica

$$\begin{aligned} d(x_1, x_2) &= d(g(x_0), g(x_1)) \leq k\ d(x_0, x_1) \\ d(x_2, x_3) &= d(g(x_1), g(x_2)) \leq k\ d(x_1, x_2) \leq k^2 d(x_0, x_1) \\ &\vdots \\ d(x_n, x_{n+1}) &= d(g(x_n), g(x_{n+1})) \leq k^n d(x_0, x_1) \end{aligned}$$

De estas desigualdades, aplicando la desigualdad triangular de las distancias, resultará que:

$$\begin{aligned} d(x_n, x_{n+p}) &\leq d(x_n, x_{n+1}) + d(x_{n+1}, x_{n+p}) \\ &\leq d(x_n, x_{n+1}) + d(x_{n+1}, x_{n+2}) + d(x_{n+2}, x_{n+p}) \\ &\leq d(x_n, x_{n+1}) + d(x_{n+1}, x_{n+2}) + \cdots + d(x_{n+p-1}, x_{n+p}) \\ &\leq k^n d(x_0, x_1) + k^{n+1} d(x_0, x_1) + \cdots + k^{n+p-1} d(x_0, x_1) \\ &= k^n d(x_0, x_1) \left[1 + k + k^2 + \cdots + k^{p-1}\right] \end{aligned}$$

En la expresión anterior, el sumatorio que aparece representa la suma de una progresión geométrica cuyo primer término toma el valor 1 y de razón k. Por tanto $\displaystyle\sum_{i=0}^{\infty} k^i = \frac{1}{1-k}$ lo que conduce a

$$d(x_n, x_{n+p}) \leq \frac{k^n}{1-k} d(x_0, x_1)$$

y puesto que, al ser $g(x)$ una contracción, k es estrictamente inferior a 1, para cualquier valor de ε positivo y bastaría considerar el índice natural N de forma que

$$N \geq \frac{\log\left(\dfrac{\varepsilon(1-k)}{d(x_0, x_1)}\right)}{\log k}$$

para que se verifique que $d(x_n, x_m) < \varepsilon$ para todo par de índices n y m mayores que N. En definitiva, la sucesión $\{x_i\}_{i=1}^{\infty}$es una sucesión de Cauchy.

Y como por las hipótesis del teorema se está trabajando en un espacio métrico completo, admitiría límite x^*. Y puesto que al ser g una contracción, es continua, entonces se verifica que

$$g(x^*) = \lim_{i\to\infty} g(x_i) = \lim_{i\to\infty} x_{i+1} = x^*$$

Luego $g(x)$ admite un punto fijo que es el límite de la sucesión generada mediante

$$x_{i+1} = g\left(x_i\right) \quad , \quad i = 0, 1, 2, \cdots$$

a partir de cualquier elemento x_0 perteneciente a E.

Demostremos ahora la unicidad del punto fijo, mediante reducción al absurdo. En efecto, consideremos por un momento que en las condiciones del teorema hubiera dos elementos distintos de E, que denotaremos por a y b, que fuesen puntos fijos. Al ser distintos se tendrá que $d(a, b) > 0$. Pero por otra parte se debe verificar que

$$d\left(a, b\right) = d\left(g\left(a\right), g\left(b\right)\right) \leq kd\left(a, b\right) < d\left(a, b\right)$$

Y que un número real sea estrictamente menor que sı mismo obviamente es absurdo.

El teorema anterior, no imposibilita que otras aplicaciones que no sean contracciones, o que estén definidas en espacios que no sean completos, puedan tener uno o varios puntos fijos. Simplemente nos asegura que si nuestra aplicación es una contracción y está definida sobre un espacio métrico completo siempre existirá un único punto fijo de la aplicación. La demostración de la existencia del punto fijo nos indica que podemos encontrarlo como límite de la sucesión

$$x(i+1) = g\left(x_i\right)$$

generada a partir de cualquier x_0 perteneciente al espacio E.

La demostración de la existencia del punto fijo que se ha realizado en el teorema precedente ya pone de manifiesto muchos aspectos importantes sobre los métodos iterativos de resolución de ecuaciones no lineales. En efecto, si logramos definir una contracción g con la que generar una sucesión que converja hacia la solución de las ecuaciones no lineales a resolver ya habremos dado un primer paso. La distancia que separe x_n de la solución $x^* = (x_\infty)$ podría estimarse mediante

$$\begin{aligned}
d\left(x_n, x^*\right) &= d\left(x_n, x_\infty\right) \leq d\left(x_n, x_{n+1}\right) + d\left(x_{n+1}, x_\infty\right) \\
&\leq d\left(x_n, x_{n+1}\right) + d\left(x_{n+1}, x_{n+2}\right) + d\left(x_{n+2}, x_\infty\right) \\
&\leq d\left(x_n, x_{n+1}\right) + d\left(x_{n+1}, x_{n+2}\right) + \cdots + d\left(x_{n+j}, x_{n+j+1}\right) + \cdots \\
&\leq k^n d\left(x_0, x_1\right) + k^{n+1} d\left(x_0, x_1\right) + \cdots + k^{n+j} d\left(x_0, x_1\right) + \cdots \\
&= k^n d\left(x_0, x_1\right) \left[1 + k + k^2 + \cdots + k^{p-1}\right] \\
&= k^n d\left(x_0, x_1\right) \sum_{i=0}^{\infty} k^i \\
&= \frac{k^n}{1-k} d\left(x_0, x_1\right)
\end{aligned}$$

Ello pone de manifiesto diferentes hechos (para el caso en que ya se haya resuelto el problema de que la sucesión converja). Entre ellos podemos citar:

1. No es indiferente el elemento x_0 con el que se inicialice la sucesión pues el "ahorrarnos" iteraciones en el proceso depende de $d(x_0, x_1)$.

2. Cuanto más próxima a 1 sea la constante de Lipschitz de la contracción más pequeño sería $(1-k)$ y por tanto mayor será el número (n) de iteraciones que se necesitarán para estar "razonablemente" cerca de la solución. En otros términos cuanto más próxima a 0 sea la constante de Lipschitz de las contracciones con las que se trabaje, menor será el esfuerzo de cálculo necesario para obtener "buenas" aproximaciones de las soluciones.

3. Si en lugar de acotar la distancia a la solución con $d(x_0, x_1)$ se acotara con $d(x_n, x_{n+1})$ se tiene que

$$\begin{aligned}
d(x_n, x^*) &= d(x_n, x_\infty) \leq d(x_n, x_{n+1}) + d(x_{n+1}, x_\infty) \\
&\leq d(x_n, x_{n+1}) + d(x_{n+1}, x_{n+2}) + d(x_{n+2}, x_\infty) \\
&\leq d(x_n, x_{n+1}) + d(x_{n+1}, x_{n+2}) + \cdots + d(x_{n+j}, x_{n+j+1}) + \cdots \\
&\leq kd(x_{n-1}, x_n) + k^2 d(x_{n-1}, x_n) + \cdots + k^j d(x_{n-1}, x_n) + \cdots \\
&= kd(x_{n-1}, x_n)\left[1 + k + k^2 + \cdots + k^j + \cdots\right] \\
&= kd(x_{n-1}, x_n) \sum_{i=0}^{\infty} k^i \\
&= \frac{k}{1-k} d(x_{n-1}, x_n)
\end{aligned}$$

Ello nos indica que la distancia entre dos aproximaciones consecutivas de la solución es una forma de medir la distancia de la última de ellas a la solución ponderada por el factor $\frac{k}{1-k}$(lo que nos vuelve a llevar a la consideración de que interesan valores de la constante de Lipschitz cuanto más pequeños mejor).

Definición 7 *El número $\alpha \in \mathbb{R}$ se dice una solución de la ecuación si se verifica que $f(\alpha) = 0$, es decir, si α es una raíz de la ecuación f.*

Definición 8 *El error absoluto, representado por e_n, se obtiene como el valor absoluto de la diferencia entre el valor exacto x y la aproximación a este valor α. Esto es*

$$e_n = |x - \alpha|$$

Definición 9 *Supongamos que la sucesión $x_n \to \alpha$, $x_n \neq \alpha$. Decimos que la sucesión converge a α con orden de convergencia $p > 0$, si $\exists$ una constante $\lambda > 0$ tal que*

$$\lim_{j\to\infty} \frac{|e_{j+1}|}{|e_j|^p} = \lim_{j\to\infty} \frac{|x_{j+1}-\alpha|}{|x_j-\alpha|^p} = \lambda$$

la constante λ se llama constante asintótica del error.

De aqui podemos inferir los siguientes resultados:

1. Si $p = 1$, la convergencia se llama lineal y, para j suficiente grande, es $|e_{j+1}| \approx \lambda |e_j|$. Para este caso, la constante λ se llama tasa de convergencia o constante asintótica y necesariamente $0 < \lambda < 1$.

2. Si $1 < p < 2$, la convergencia se llama superlineal.

3. Si $p = 2$, la convergencia se llama cuadrática y, para j suficiente grande, se cumple $|e_{j+1}| \approx \lambda |e_j|^2$. Si una sucesión tiene convergencia cuadrática, a partir de un cierto momento, el número de decimales exactos se duplica a cada paso.

4. Si $p = 3$, la convergencia se llama cúbica.

5. En general, la convergencia es de orden p, si existen $\lambda > 0$ y k_0 tales que $|x_{k+1} - \alpha| \leq \lambda\, |x_{k+1} - \alpha|^p\,, \quad \forall\, k \geq k_0$

También se puede definir al orden de convergencia de la sucesión $\{x_n\}_{n=0}^{\infty}$ como el número $p \geq 1$ tal que

$$|e_{n+1}| \leq C\, |e_n|^p \quad \text{para algún } C > 0 \quad y \quad 0 < C < 1$$

En general, una sucesión con mayor orden de convergencia se aproxima a la raíz más rápidamente que una de orden inferior. La constante asintótica influye en la rapidez de convergencia, pero no es tan importante como el orden. A una técnica iterativa que produce una sucesión que converge con orden de convergencia p se le denomina, en general, método de orden p.

Las nociones de orden de convergencia y de constante asintótica del error tienen una gran importancia práctica. Cuando x_n está suficientemente cerca de α, entonces

$$|x_{n+1} - \alpha| \approx \lambda\, |x_n - \alpha|^p$$

y haciendo

$$d_n = -\log |x_n - \alpha|$$

obtenemos

$$d_{n+1} = pd_n + r$$

con

$$r = -\log \lambda$$

Esto quiere decir que en cada iteración, se multiplica por p el número de cifras exactas, de forma que, en general, un método con un orden de convergencia alto converge más rápidamente que un algoritmo de orden más bajo. La constante asintótica también afecta a la rapidez de convergencia, pero no es comparativamente tan importante como el orden.

Teorema 6.3 *Sea $\{x_n\}_{n=0}^{\infty}$ una sucesión que converge a α y $e_n = x_n - \alpha$, $p \geq 1$, $k_i \in \mathbb{R}$, $k_i \neq 0$, $i = 0, 1, 2, \cdots$. Si $e_{n+1} = k_1 e_n^p + k_2 e_n^{p+1} + \cdots$ o $e_{n+1} = k_1 e_n^p + O\left(e_n^{p+1}\right)$, entonces*

$$\lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n|^p} = |k_1|$$

Demostración 3 *Sea $\{x_n\}_{n=0}^{\infty}$ una sucesión que converge a α y $e_n = x_n - \alpha$, $p \geq 1$, $k_i \in \mathbb{R}$, $k_i \neq 0$, $i = 0, 1, 2, \cdots$. Por hipótesis*

$$e_{n+1} = k_1 e_n^p + k_2 e_n^{p+1} + k_3 e_n^{p+2} + \cdots$$

tomando e_n^p factor común, obtenemos

$$e_{n+1} = e_n^p \left(k_1 + k_2 e_n^1 + k_3 e_n^2 + \cdots\right)$$

asi

$$\frac{e_{n+1}}{e_n^p} = k_1 + k_2 e_n^1 + k_3 e_n^2 + \cdots$$

como $e_n \to 0$ cuando $n \to \infty$, se tiene

$$\lim_{n\to\infty} \frac{|e_{n+1}|}{|e_n|^p} = |k_1|$$

Definición 10 *Se dice que el cero α de la función f tiene multiplicidad m $(m \in \{1, 2, 3, ...\})$ si la función f se puede escribir en la forma*

$$f(x) = (x - \alpha)^m g(x) \quad , \quad x \neq \alpha$$

donde $\lim_{x\to\alpha} g(\alpha) \neq 0$.

Si $m = 1$, se dice que α es una raíz simple; si $m > 1$, se dice que α es una raíz no simple o raíz múltiple.

Teorema 6.4 *Sea $f \in C^1[a,b]$ y sea $\alpha \in (a,b)$. Entonces f tiene un cero simple (de multiplicidad 1) en α si, y solo si*

$$f(\alpha) = 0 \quad y \quad f'(\alpha) \neq 0$$

Sea $f \in C^m[a,b]$ y sea $p \in (a,b)$. Entonces α es un cero de f de multiplicidad m si y solo si

$$f(\alpha) = 0\ ,\quad f'(\alpha) = 0\ , \cdots ,\quad f^{(m-1)}(\alpha) = 0 \quad y \quad f^{(m)}(\alpha) \neq 0$$

Demostración 4 *Por definición*

$$f(x) = (x-\alpha)^m g(x) \quad , \quad x \neq \alpha$$

aplicando la fórmula para la derivada de un producto

$$\begin{aligned} f^{(j)}(x) &= \sum_{k=0}^{j} \binom{j}{k} \left[(x-\alpha)^m\right]^{(k)} g(x)^{(j-k)} \\ &= \sum_{k=0}^{j} \binom{j}{k} m(m-1)\cdots(m-k+1)(x-\alpha)^{(m-k)} g(x)^{(j-k)} \end{aligned}$$

si $j < m$. se tiene que $f^{(j)}(\alpha) = 0$ y si $j = m$, $f^{(m)}(\alpha) = m!$.

Recíprocamente, desarrollando la serie de Taylor

$$f(x) = f(\alpha) + f'(\alpha)(x-\alpha) + \frac{f''(\alpha)}{2}(x-\alpha)^2 + \cdots + \frac{f^{(p-1)}(\alpha)}{(p-1)!}(x-\alpha)^{p-1} + \frac{f^{(p)}(\xi)}{p!}(x-\alpha)^p\ , \quad x < \xi < \alpha$$

aplicando la hipótesis, obtenemos

$$f(x) = \frac{f^{(p)}(\xi)}{p!}(x-\alpha)^p\ , \quad x < \xi < \alpha$$

siendo $f^{(p)}(x)$ continua en α,

$$f(x) = (x-\alpha)^p g(x)$$

con $g(x) = \dfrac{f^{(p)}(\xi)}{p!}$, $\quad \xi = \xi(x)$

Definición 11 *Si r es el orden de convergencia del método iterativo y d es el número de evaluaciones funcionales por iteración de dicho método, entonces, la eficiencia computacional está dada por*

$$\rho = r^{1/d} = \sqrt[d]{r}$$

En general, una sucesión con mayor eficiencia computacional es mejor que otra sucesión de menor eficiencia computacional.

Definición 12 *Sea α una raíz de la ecuación $f(x) = 0$ y supongamos que x_{n+1} , x_n y x_{n-1} son tres iteraciones consecutivas cercanas a la raíz α. El orden de convergencia computacional ρ puede ser aproximado por la fórmula*

$$\rho = \frac{\ln \left| \dfrac{x_{n+1} - \alpha}{x_n - \alpha} \right|}{\ln \left| \dfrac{x_n - \alpha}{x_{n-1} - \alpha} \right|}$$

Definición 13 *Dada una sucesión $x_n \to x$ y una sucesión $y_n \to y$, diremos que (y_n) converge más rapidamente que (x_n) si*

$$\lim_{n \to \infty} \frac{y - y_n}{x - x_n} = 0$$

Por otro lado, si dos sucesiones tienen el mismo orden de convergencia y la misma constante asintótica, no tienen porqué converger con la misma velocidad.

Por ejemplo:

$$y_n = \frac{a^n}{n^2} \quad , \quad x_n = \frac{a^n}{n} \quad , \quad 0 < a \leq 1$$

el orden es 1 y la constante asintótica es a para ambas, pero (y_n) converge más rápidamente que (x_n).

Fórmula de Taylor

Si $f \in C^{n+1}[a,b]$ y $x_0 \in [a,b]$, entonces para cada $x \in [a,b]$

$$f(x) = \sum_{k=0}^{n} \frac{f^{(k)}(x-h)}{k!} (x - x_0)^k + O\left((x - x_0)^{n+1}\right)$$

donde $h = x - x_0$

Criterios de Parada.

Resulta común que un proceso iterativo se detenga ya sea porque alcanzó un máximo número de iteraciones o porque logró alcanzar una tolerancia del error (prescrita por el usuario) entre la solución aproximada y la solución exacta del problema. Para este último propósito existen criterios de parada que dependen tanto de la multiplicidad de la raíz como del orden de convergencia de la función de iteración.

Debido a que los métodos a ser analizados están propuestos para ceros simples y sus órdenes de convergencia son mayores que uno, dos criterios de parada son implementados: para una tolerancia ε del error se pide que

$$|f(x_n)| \leq \varepsilon \qquad o \qquad |x_{n+1} - x_n| \leq \varepsilon$$

A continuación se analizan ambos criterios.

Criterio 1: $(|f(x_n)| \leq \varepsilon)$

Supongamos que f es diferenciable en algún intervalo $I \subset \mathbb{R}$ que contenga α, considere que $x_n \in I$, $\forall\ n$. Sea x_n la n-ésima aproximación a α, por el Teorema del Valor Intermedio se tiene

$$f(x_n) = f(x_n) - f(\alpha) = f'(\xi_n)(x_n - \alpha) \quad con \quad \xi_n \in (x_n, \alpha)$$

por tanto

$$x_n - \alpha = \frac{f(x_n)}{f'(\xi_n)}$$

de la fórmula anterior, se desprende que:

1. $|f'(\alpha)| \approx 1$ entonces $|x_n - \alpha| \approx |f(x_n)| \leq \varepsilon$
2. $|f'(\alpha)| \lll 1$ entonces $|x_n - \alpha| \ggg \varepsilon$
3. $|f'(\alpha)| \ggg 1$ entonces $|x_n - \alpha| \lll \varepsilon$

Consecuentemente, este criterio de parada no funciona para raíces múltiples, pues si $f'(\alpha) = 0$ entonces se cumple el caso (2.), lo cual produciría una parada del proceso iterativo sin haber alcanzado la tolerancia prescrita. En cambio, para ceros simples, el criterio funciona si se cumple el caso (1.) ó el caso (3.); aunque, para este último se estaría haciendo iteraciones innecesarias.

Criterio 2: $(|x_{n+1} - x_n| \leq \varepsilon)$

Sea $x_{n+1} = g(x_n)$ y $x_n \approx \alpha$, aplicando el teorema del valor intermedio se tiene que existe $\xi_n \in (x_n, \alpha)$ tal que

$$x_{n+1} - \alpha = g(x_n) - g(\alpha) = g'(\xi_n)(x_n - \alpha) \approx g(\alpha)(x_n - \alpha)$$

de donde

$$\begin{aligned} x_{n+1} - x_n &= (x_{n+1} - \alpha) - (x_n - \alpha) \\ &\approx g'(\alpha)(x_n - \alpha) - (x_n - \alpha) \\ &= (x_n - \alpha)(g'(\alpha) - 1) \end{aligned}$$

por tanto

$$x_n - \alpha \approx \frac{x_{n+1} - x_n}{g'(\alpha) - 1}$$

Luego, si $g'(\alpha) \approx 1$, entonces $|x_n - \alpha| \geq |x_{n+1} - x_n|$ para este caso podría no ser un buen criterio para aquellos métodos que presentan una convergencia lineal, pues $g'(\alpha) \neq 0$. Si $-1 < g'(\alpha) < 0$ entonces $|x_n - \alpha| \leq |x_{n+1} - x_n|$, por lo tanto, el criterio es bueno.

6.4. Método del Punto Fijo

El método de aproximaciones sucesivas o iteración de punto fijo es una forma muy útil y simple de encontrar la raíz de una ecuación de la forma $f(x) = 0$. Para ello se reordena la ecuación de manera que x sea igual a $g(x)$. Esta transformación se puede llevar a cabo mediante operaciones algebraicas o simplemente agregando x en ambos miembros de la ecuación original. A una solución de esta ecuación se le llama un punto fijo de la función g. Sin embargo, es muy importante la selección de la función $g(x)$, ya que no siempre converge con cualquier forma elegida de $g(x)$.

Conexión entre la búsqueda de puntos fijos y búsqueda de raíces

Si g tiene punto fijo p, entonces $f(x) = g(x) - x$ tiene un cero en p. Si f tiene una raíz p, entonces $g(x) = x - f(x)$ tiene punto fijo p.

Teorema 6.5 *Sea $g : [a; b] \to \mathbb{R}$ una función diferenciable que verifica*

1. $g([a, b]) \subseteq (a, b)$

2. $\max\limits_{x \in [a,b]} |g'(x)| < 1$

Entonces existe un único $p \in [a; b]$ tal que $g(p) = p$ y además para todo $x_0 \in [a; b]$, la sucesión $\{x_n\}$ generada por la iteración $x_{n+1} = g(x_n)$ converge a p

Demostración 5 *Sea $h(x) = g(x) - x$, $\forall x \in [a, b]$. Entonces $h(x)$ es continua y verifica que $h(a) > 0$ y $h(b) < 0$, por lo que se verifican las condiciones del Teorema de Bolzano. En consecuencia, existe un $p \in (a, b)$ tal que $h(p) = 0$, es decir $g(p) = p$.*

Para ver la unicidad, supongamos que existen dos valores, $p, t \in (a, b)$ tales que $g(p) = p$ y $g(t) = t$, entonces por el teorema del valor medio existe $c \in (p, t)$ tal que $g(t) - g(p) = g'(c)(t - p)$, es decir, $g'(c) = 1$, lo que contradice la segunda condición.

Sea ahora $\{x_n\}$ la sucesión generada a partir de $x_0 \in [a, b]$ mediante la iteración $x_{n+1} = g(x_n) \quad , \quad n \geq 0$ y $L = \max\limits_{x \in [a,b]} |g'(x)| < 1$. Se verifica entonces que

$$e_n = |x_n - p| = |g(x_{n-1}) - g(p)| = |g'(x)| \, |e_{n-1}| \leq Le_{n-1} \leq L^2 e_{n-2} \leq \cdots \leq L^n e_0 \qquad \blacksquare$$

Teorema 6.6 *Sea $g : I \subset [a; b] \to R$ una función continua en $[a, b]$. Si (x_n) es una sucesión en $[a, b]$ que converge a $\alpha \in [a, b]$, entonces α es un punto fijo de g.*

Demostración 6 *Como $\lim\limits_{n\to\infty} (x_n) = \alpha$, se tiene que $\lim\limits_{n\to\infty} (x_{n+1}) = \alpha$. A su vez, como se tiene que, g es continua en $[a, b]$ y por la relación $x_{n+1} = g(x_n)$, se obtiene*

$$g(\alpha) = g\left(\lim_{n\to\infty} (x_n)\right) = \lim_{n\to\infty} g(x_n) = \lim_{n\to\infty} x_{n+1} = \alpha$$

de donde α es el punto fijo de g. $\blacksquare$

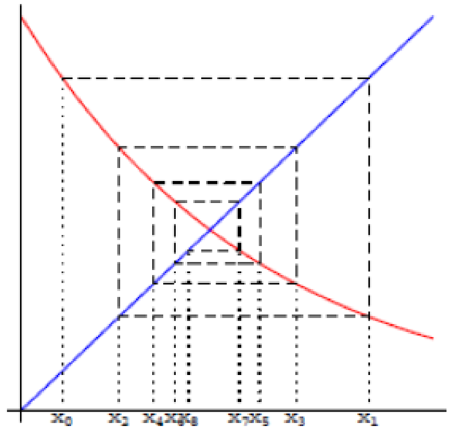
Para aproximar el punto fijo de una función g, se escoge una aproximación inicial x_0 y se genera la sucesión (x_n) dada por $x_{n+1} = g(x_n)$ para cada $n > 0$. Esta técnica recibe el nombre de iteración de punto fijo.

En general, el algoritmo de un método iterativo está dado por:

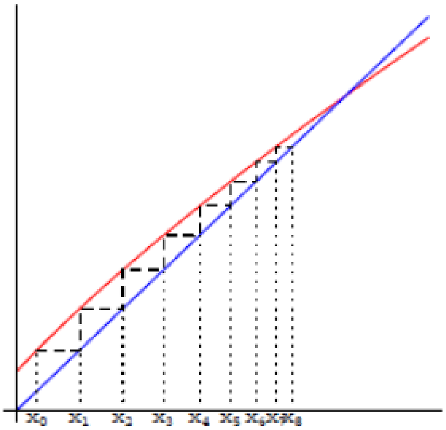
1. Dado $x_0 \in [a, b]$
2. Para $n = 1, 2, \cdots$, hacer $x_{n+1} = g(x_n)$

Interpretación gráfica de los métodos iterativos.

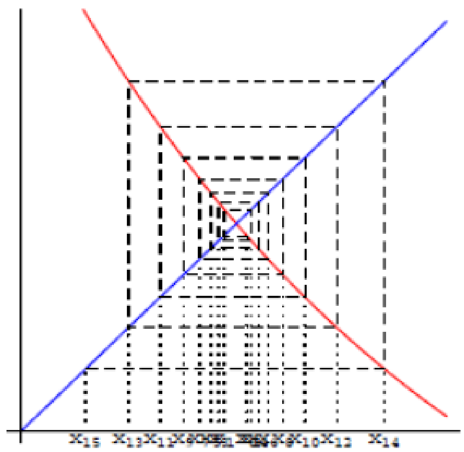
Según sea $g(x)$ y se elija x_0, los métodos iterativos pueden ser convergentes o divergentes y, en ambos casos pueden variar en forma espiral o en escalera, como se indican en los siguientes gráficos.



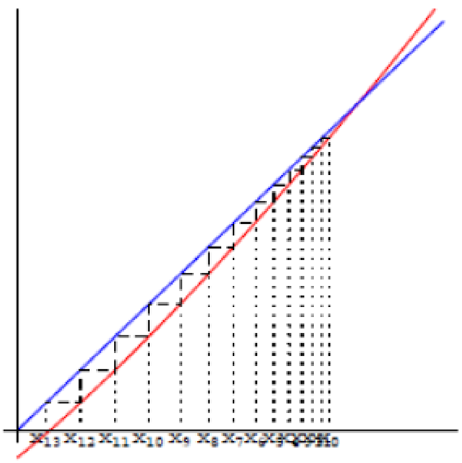
Convergencia oscilante o en espiral



Convergencia monótona o en escalera



Divergencia oscilante o en espiral



Divergencia monótona o en escalera

6.5. Métodos Iterativos

Encontrar una o más raíces de una ecuación no lineal de la forma, $f(x) = 0$, donde $f : I \to \mathbb{R}$ es una función real de variable real, es uno de los problemas más comunes que ocurren en las matemáticas aplicadas. Los métodos analíticos para resolver tales problemas son difíciles de encontrar o en línea general no existen. Por lo tanto, para resolver este problema resulta necesario aplicar técnicas numéricas basadas en procedimientos iterativos como por ejemplo, Newton-Raphson.

En los últimos años se han definido nuevos métodos iterativos que mejoran, en cierta forma, la precisión de los métodos clásicos.

A lo largo de este trabajo se considera la ecuación no lineal $f(x) = 0$ donde f es una función real de variable real y se asume que α es una raíz simple de f.

Teoría General de los Métodos Iterativos

El método iterativo de Newton pueden considerarse un caso particular del siguiente método iterativo más general: sea x_{n+1} determinado por evaluaciones de la función y/o de las derivadas en los puntos $x_n,\ x_{n-1},\ x_{n-2},\ \cdots, x_{n-m+1}$ y hagamos

$$x_{n+1} = g\left(x_n, x_{n-1},\ x_{n-2},\ \cdots, x_{n-m+1}\right) \quad , \quad n = m,\ m+1,\ \cdots$$

La función g se llama función de iteración, y $\{x_n\}_{n=0}^{\infty}$ la sucesión de iterados.

En los métodos presentado en la sección anterior, se tiene

$$\begin{aligned} g\left(x\right) &= x - \frac{f\left(x\right)}{f'\left(x\right)} \quad , \quad m = 1 \quad , \quad \textit{para Newton} \\ g\left(x, y\right) &= x - f\left(x\right)\frac{x-y}{f\left(x\right) - f\left(y\right)} \quad , \quad m = 2 \quad , \quad \textit{para la Secante} \end{aligned}$$

La teoría general de los métodos iterativos es simple cuando $m = 1$, es decir,

$$x_{n+1} = g(x_n), \qquad n = 0, 1, 2, ...$$

Sea $\{x_n\}_{n=0}^{\infty}$ una sucesión generada para un x_0 inicial dado. Supongamos que $x_n \to \alpha$ y que g es continua. Tomando límite

$$\alpha = g\left(\alpha\right)$$

Si esto es cierto decimos que α es un punto fijo de g. Para resolver el problema $f(x) = 0$, se puede construir una función g tal que un punto fijo de g sea una raíz de f.

Teorema 6.7 *Si $g \in C[a, b]$ y $g : [a, b] \to [a, b]$, entonces g tiene un punto fijo en $[a, b]$.*

Demostración 7 *Si $g(a) = a$ o si $g(b) = b$, entonces g tendrá un punto fijo en un extremo. Supongamos que no es así; entonces deberá ser cierto que $g(a) > a$ y que $g(b) < b$. La función $h(x) = g(x) - x$ es continua en $[a, b]$ y tenemos*

$$\begin{aligned} h(a) &= g(a) - a > 0 \\ & \quad y \\ h(b) &= g(b) - b < 0 \end{aligned}$$

Por el Teorema del Valor Intermedio se tiene que existe $\alpha \in (a,b)$ para el cual $h(\alpha) = 0$. Este número α es un punto fijo de g.

$$0 = h(\alpha) = g(\alpha) - \alpha \;\; implica \;\; que \;\; g(\alpha) = \alpha \qquad \blacksquare$$

Teorema 6.8 *Sea g continuamente diferenciable en algún intervalo (c,d) que contenga el punto fijo α de g. Si $|g\prime(\alpha)| < 1$, entonces existe una $\delta > 0$ tal que la iteración de punto fijo converge para cualquier aproximación x_0 siempre que $|x_0 - \alpha| \leq \delta$.*

Demostración 8 *Por ser $g\prime$ continua y $|g\prime(\alpha)| < 1$, entonces existe $\delta >$ 0 tal que $|g\prime(x)| < 1$ para todo $x \in [\alpha - \delta, \alpha + \delta]$.*

Para continuar la prueba es necesario demostrar que g envía $[\alpha - \delta, \alpha + \delta]$ en $[\alpha - \delta, \alpha + \delta]$. Si $x \in [\alpha - \delta, \alpha + \delta]$, el Teorema del Valor Medio implica que, para algún número ξ entre x y α, $|g(x) - g(\alpha)| = |g\prime(\xi)||x - \alpha|$. Por tanto,

$$|g(x) - \alpha| = |g(x) - g(\alpha)| = |g\prime(\xi)||x - \alpha| < |x - \alpha|$$

Puesto que $x \in [\alpha - \delta, \alpha + \delta]$, se tiene que $|x - \alpha| \leq \delta$ y que $|g(x) - \alpha| < \delta$.

Este resultado implica que g manda $[\alpha - \delta, \alpha + \delta]$ en $[\alpha - \delta, \alpha + \delta]$, entonces la iteración definida por

$$x_n = g(x_{n-1}), n \geq 1$$

converge para cualquier aproximación $x_0 \in [\alpha - \delta, \alpha + \delta]$, con lo cual queda demostrado dicho teorema. $\blacksquare$

Los siguientes resultados teóricos serán de utilidad en el análisis del orden de los métodos iterativos a estudiar.

Teorema 6.9 *Sea $f \in C^3$, $h \in C^2$ en un entorno de una raíz simple de f. Si $h(\alpha) = 1$ y $h\prime(\alpha) = \dfrac{1}{2}\dfrac{f''(\alpha)}{f'(\alpha)}$, entonces existe $\delta > 0$ tal que $\forall\ x_0 \in [\alpha - \delta, \alpha + \delta]$ el método iterativo definido por $\Phi(x_n) = x_n - h(x_n)$ converge a α y además es de al menos orden tres.*

Demostración 9 *Para demostrar que Φ es convergente en un entorno de α tenemos que verificar que $\Phi(\alpha) = \alpha$ y $|\Phi\prime(\alpha)| < 1$*

$$\begin{aligned}
\Phi(\alpha) &= \alpha - h(\alpha)\frac{f(\alpha)}{f'(\alpha)} = \alpha \\
\Phi'(\alpha) &= 1 - h'(\alpha)\frac{f(\alpha)}{f'(\alpha)} - h(\alpha)\left[\frac{[f'(\alpha)]^2 - f''(\alpha)f(\alpha)}{[f'(\alpha)]^2}\right] \\
&= 1 - \frac{f'(\alpha)^2}{f'(\alpha)^2} = 1 - 1 = 0 < 1
\end{aligned}$$

De las ecuaciones anteriores y el teorema anterior, existe $\delta > 0$ tal que Φ converge a α para cualquier aproximación inicial $x_0 \in [\alpha - \delta, \alpha + \delta]$.

Derivando nuevamente, se obtiene

$$\begin{aligned}
\Phi'(\alpha) &= -\left[2h'(\alpha) + h(\alpha)\left(\frac{f'(\alpha)^3 f''(\alpha) - 2f'(\alpha)^3 f''(\alpha)}{f'(\alpha)^4}\right)\right] \\
&= -\left[2h'(\alpha) - \frac{f'(\alpha)^3 f''(\alpha)}{f'(\alpha)^4}\right] \\
&= -\left[2\frac{1}{2}\frac{f''(\alpha)}{f'(\alpha)} - \frac{f''(\alpha)}{f'(\alpha)}\right] = 0
\end{aligned}$$

de donde deducimos que Φ es de al menos orden tres. ■

Teorema 6.10 *Sea α una raíz simple de $f \in C^3(I)$, con I un intervalo abierto que contiene a α. Supongamos que $h \in C^2(I)$. Si $F(x) = xe^{-\frac{f(x)}{xf'(x)}}$ con $\phi(\alpha) = \alpha$, $\phi'(\alpha) = 0$, $\phi''(\alpha) = 0$, $\alpha \neq 0$ y $h(x) = 1 + \frac{1}{2}F'(x) + \ln\sqrt{\left|\frac{\phi(x)}{x}\right|}$, entonces existe $\delta > 0$ tal que para todo $x_0 \in [\alpha - \delta, \alpha + \delta]$ el método iterativo definido por $\Phi(x_n) = x_n - h(x_n)$ converge a α y tiene por lo menos un tercer orden de convergencia.*

Demostración 10 *La idea es verificar que $h(\alpha) = 1$ y $h'(\alpha) = \frac{1}{2}\frac{f''(\alpha)}{f'(\alpha)}$. Realizando los cálculos de $F'(x)$ y $F''(x)$ evaluada en α, se tiene*

$$\begin{aligned}
F'(x) &= 0 \\
F''(x) &= \frac{1}{\alpha}\frac{f''(\alpha)}{f'(\alpha)}
\end{aligned}$$

y al evaluar en α la derivada de $\ln\sqrt{\left|\dfrac{\phi(x)}{x}\right|}$ se obtiene $-\dfrac{1}{2\alpha}$

Asi

$$\begin{aligned} h(\alpha) &= 1+\frac{1}{2}F'(\alpha)+\ln\sqrt{\left|\frac{\phi(\alpha)}{\alpha}\right|}=1 \\ h'(\alpha) &= \frac{1}{2}\left(\frac{1}{\alpha}+\frac{f''(\alpha)}{f'(\alpha)}\right)-\frac{1}{2\alpha}=\frac{1}{2}\frac{f''(\alpha)}{f'(\alpha)} \end{aligned}$$

y por el teorema anterior, existe $\delta>0$ tal que para todo $x_0\in[\alpha-\delta,\alpha+\delta]$ el método iterativo definido por

$$\Phi(x_n)=x_n-h(x_n)\frac{f(x_n)}{f'(x_n)}$$

converge a α y tiene por lo menos un tercer orden de convergencia. ■

7. CAPÍTULO II : MÉTODO DE NEWTON

Se sabe que el esquema iterativo del método de Newton está definido por

$$x_{n+1} = x_n - \frac{h(x_n)}{h'(x_n)}$$

donde la función $h(x)$ es diferenciable y además $h'(x) \neq 0$. Este método posee orden de convergencia cuadrática y eficiencia computacional $\sqrt{2}$. El principal objetivo de este capítulo es generar algunas variantes del algoritmo de Newton y así obtener nuevos y diferentes esquemas iterativos. La generación de estas variantes radica en la sustitución de la función $h(x)$ por otra forma funcional que cumpla con las hipótesis del método de Newton y de esta manera, obtener nuevos esquemas y nuevos métodos iterativos.

En este capítulo se utilizará el desarrollo de Taylor para deducir el método de Newton.

Desarrollemos la función $h(x)$ mediante el desarrollo de Taylor de orden 2, alrededor del punto $x = a$, entonces

$$h(x) = h(a) + h'(a)(x-a) + O\left(h^2\right)$$

como $h(x) = 0$ se tiene

$$\begin{aligned} 0 &= h(a) + h'(a)x - h'(a)a \\ h'(a)x &= h'(a)a - h(a) \\ x &= \frac{h'(a)a - h(a)}{h'(a)} \\ x &= a - \frac{h(a)}{h'(a)} \end{aligned}$$

luego, el esquema iterativo toma la forma

$$x_{n+1} = x_n - \frac{h(x_n)}{h'(x_n)}$$

7.1. Análisis de Convergencia del Método de Newton

A partir de la fórmula iterativa

$$f(x_k) + f'(x_k)(x_{k+1} - x_k) = 0 \quad , \quad \forall\, k \geq 0$$

ahora, utilizando el desarrollo de Taylor de f de orden dos

$$0 = f(\alpha_k) = f(x_k) + f'(x_k)(\alpha_k - x_k) + \frac{1}{2}f''(\alpha_k)(\alpha_k - x_k)^2 \quad , \quad \forall\, k \geq 0$$

donde α_k es un punto situado entre α y x_k.

Restando las dos últimas expresiones, resulta

$$f'(x_k)(\alpha_k - x_k) + \frac{1}{2}f''(\alpha_k)(\alpha_k - x_k)^2 = 0 \quad , \quad \forall\, k \geq 0$$

si

$$p = \min_{x\in[a,b]} |f'(x)| \quad y \quad q = \max_{x\in[a,b]} |f''(x)|$$

de la expresión anterior obtenemos

$$|x_k - \alpha| \leq \frac{q}{2p}|x_k - \alpha|^2 \quad , \quad \forall\, k \geq 0$$

con lo cual, queda demostrado que el orden de convergencia del método de Newton-Raphson es de orden cuadrático. ■

Si se supone que hay convergencia, entonces x_r se debe aproximar a la raíz real x y la magnitud del error es proporcional al cuadrado del error anterior. Esto significa que el número de cifras decimales correctas aproximadamente se duplica en cada iteración. A este comportamiento se le llama convergencia cuadrática.

Ahora vamos a considerar algunos casos especiales donde la función $h(x)$ puede expresarse como combinación lineal de las funciones $f(x)$ y $g(x)$. En todos los casos, vamos a tomar la función $g(x)$ como una función arbitraria auxiliar y $\alpha \in \mathbb{R}$.

7.2. Variantes del Método de Newton

7.2.1. Caso I : Forma funcional $h(x) = f(x) + g(x)$

Consideremos la función $h(x)$ con la forma funcional de una suma, es decir

$$h(x_n) = f(x_n) + g(x_n)$$

donde $f(x_n)$ y $g(x_n)$ son funciones diferenciables y $f'(x_n) \neq -g'(x_n)$.

Sustituyendo en el esquema anterior $h'(x) = f'(x) + g'(x)$, obtenemos el nuevo esquema iterativo

$$x_{n+1} = x_n - \frac{f(x_n) + g(x_n)}{f'(x_n) + g'(x_n)}$$

A partir de la formulación de este nuevo esquema iterativo, vamos a considerar a la función $g(x_n)$ como una función auxiliar arbitraria que puede tomar diferentes formas funcionales y a partir de aquí, obtener nuevas variantes del método de Newton:

Consideremos la función $g(x_n) = \alpha f(x_n)$, al derivar y sustituir en el esquema iterativo anterior, obtenemos una variante del método de Newton, el cual se presenta en la siguiente fórmula

$$x_{n+1} = x_n - \frac{f(x_n) + \alpha f(x)}{f'(x_n) + \alpha f'(x_n)} \tag{1}$$

Observemos que si $\alpha = 0$ o $\alpha = 1$ se obtiene el método de Newton en su forma clásica

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Algoritmo 1 *Al considerar $g(x_n) = \alpha f^2(x_n)$, obtenemos otra variante*

$$x_{n+1} = x_n - \frac{f(x_n) + \alpha f^2(x_n)}{f'(x_n) + 2\alpha f(x_n) f'(x_n)} \tag{2}$$

Al considerar $\alpha = 0$, se obtiene el método de Newton en su forma clásica

Podemos generalizar el proceso anterior, haciendo $g(x_n) = \alpha [f(x_n)]^n$, donde $n \in \mathbb{Z}^+$, efectuando las operaciones indicadas en el esquema iterativo, obtenemos otra variante del método de Newton, es decir

$$x_{n+1} = x_n - \frac{f(x_n) + \alpha f^n(x_n)}{f'(x_n) + n\alpha [f(x_n)]^{n-1} f'(x_n)} \tag{3}$$

Notemos que al considerar $\alpha = 0$, se obtiene el método de Newton en su forma clásica. Por otro lado, para $n = 1$ y $\alpha = 1$, también obtenemos el método de Newton en su forma estándar.

En particular, para $n = 2$ y $\alpha = 2$, se obtiene el siguiente

Algoritmo 2

$$x_{n+1} = x_n - \frac{f(x_n) + 2f^2(x_n)}{f'(x_n) + 4\alpha f(x_n) f'(x_n)} \tag{4}$$

Algoritmo 3 *Sea $g(x) = f(x) e^{-\alpha x}$, entonces el método de Newton, toma la forma*

$$x_{n+1} = x - \frac{f(x) + f(x) e^{-\alpha x}}{f'(x) + f'(x) e^{-\alpha x} - \alpha f(x) e^{-\alpha x}} \tag{5}$$

Notemos que para $\alpha = 0$, el método iterativo se reduce al método clásico de Newton.

Algoritmo 4 *Sea $g(x) = f(x)e^{\alpha x}$, entonces el método de Newton, toma la forma*

$$x_{n+1} = x - \frac{f(x) + f(x)e^{\alpha x}}{f'(x) + f'(x)e^{\alpha x} + \alpha f(x)e^{\alpha x}} \tag{6}$$

Notemos que para $\alpha = 0$, el método iterativo se reduce al método clásico de Newton.

Ahora si $g(x_n) = f'(x_n)$, luego tenemos

$$x_{n+1} = x_n - \frac{f(x_n) + f'(x_n)}{f'(x_n) + f''(x_n)} \tag{7}$$

continuando con el proceso, si $g(x_n) = f''(x_n)$, entonces

$$x_{n+1} = x_n - \frac{f(x_n) + f''(x_n)}{f'(x_n) + f'''(x_n)} \tag{8}$$

Y de manera generalizada, se obtiene que para $g(x_n) = f^{(n)}(x_n)$

$$x_{n+1} = x_n - \frac{f(x_n) + f^{(n)}(x_n)}{f'(x_n) + f^{(n+1)}(x_n)} \tag{9}$$

Consideremos algunos casos donde interviene la recíproca de la funcion derivada, por ejemplo, la siguiente fórmula

Sea $g(x_n) = \dfrac{1}{f'(x_n)}$, entonces la nueva variante obtenida es

$$x_{n+1} = x_n - \frac{[f'(x_n)]^2 f(x_n) + f'(x_n)}{[f'(x_n)]^3 - f''(x_n)} \tag{10}$$

Por otro lado, sea $g(x_n) = \dfrac{1}{f''(x_n)}$, se tiene otra variante

$$x_{n+1} = x_n - \frac{f''(x_n)[f(x_n)f''(x_n) + 1]}{f'(x_n)[f''(x_n)]^2 - f'''(x_n)} \tag{11}$$

De manera más general, tenemos que si $g(x_n) = \dfrac{1}{f^{(n)}(x_n)}$, entonces obtenemos que la variante del método de Newton es

$$x_{n+1} = x_n - \frac{f^{(n)}(x_n)\left[f(x_n)f^{(n)}(x_n) + 1\right]}{f'(x_n)\left[f^{(n)}(x_n)\right]^2 - f^{(n+1)}(x_n)} \tag{12}$$

Analicemos algunos casos donde la función auxiliar $g(x_n)$ es combinación de cociente de derivadas, por ejemplo:

Sea $g(x_n) = \dfrac{f(x_n)}{f'(x_n)}$, entonces la nueva variante es

$$x_{n+1} = x_n - f(x_n)f'(x_n)\frac{f'(x_n) + 1}{[f'(x_n)]^3 - [f'(x_n)]^2 + f(x_n)f''(x_n)} \tag{13}$$

Sea $g(x_n) = \dfrac{f''(x_n)}{f'(x_n)}$, luego obtenemos

$$x_{n+1} = x_n - \frac{f'(x_n)f''(x_n) + f(x_n)[f'(x_n)]^2}{[f'(x_n)]^3 - f'(x_n)f'''(x_n) + [f''(x_n)]^2} \tag{14}$$

7.2.2. Caso II : Forma funcional $h(x) = f(x) - g(x)$

La función $h(x)$ tiene la forma funcional de una diferencia, es decir

$$h(x) = f(x) - g(x)$$

donde

$$h'(x) = f'(x) - g'(x)$$

sustituyendo en el esquema principal del método de Newton, obtenemos el nuevo esquema iterativo

$$x_{n+1} = x_n - \frac{f(x_n) - g(x_n)}{f'(x_n) - g'(x_n)}$$

Nuevamente, vamos a considerar la función $g(x_n)$ como una función auxiliar arbitraria que puede tomar diferentes formas funcionales y de esta manera, obtendremos nuevas variantes del Método de Newton.

Para iniciar con el estudio de este caso, es evidente que si $g(x) = f(x)$, entonces el método de Newton se indetermina, es decir, dicho método colapsa.

Sea $g(x) = \alpha f^2(x)$, $\alpha \in \mathbb{R}$ y nuevamente sustituyendo, obtenemos una nueva variante del método de Newton, es decir

$$x_{n+1} = x_n - \frac{f(x_n) - \alpha f^2(x_n)}{f'(x_n) - 2\alpha f(x_n) f'(x_n)} \tag{15}$$

Observemos que para $\alpha = 0$, se obtiene el método de Newton en su forma clásica.

De manera general, consideremos que $g(x) = [f(x)]^n$, $n \in \mathbb{R}$ y nuevamente sustituyendo, obtenemos otra variante del método de Newton, es decir

$$x_{n+1} = x_n - \frac{f(x_n) - f^n(x_n)}{f'(x_n) - n[f(x_n)]^{n-1} f'(x_n)} \tag{16}$$

Notemos que para $n = 1$, la variante obtenida colapsa.

Para $n = 11$, obtenemos el siguiente

Algoritmo 5

$$x_{n+1} = x_n - \frac{f(x_n) - f^{11}(x_n)}{f'(x_n) - 11 f(x_n)^{10} f'(x_n)} \tag{17}$$

Algoritmo 6 *Sea* $g(x) = \dfrac{1}{f(x)}$, *entonces el método de Newton, toma la forma*

$$x_{n+1} = x_n - \frac{[f(x_n)]^3 - f(x_n)}{f'(x_n)[f(x_n)]^2 - f'(x_n)} \tag{18}$$

Algoritmo 7 *Sea* $g(x) = \dfrac{f(x)}{f'(x)}$, *entonces el método de Newton, toma la forma*

$$x_{n+1} = x_n - \frac{f(x_n)[f'(x_n)]^2 - f(x_n) f'(x_n)}{[f'(x_n)]^3 - [f'(x_n)]^2 + f(x_n) f''(x_n)} \tag{19}$$

Algoritmo 8 *Sea* $g(x) = f(x) e^{-\alpha x}$, *entonces el método de Newton, toma la forma*

$$x_{n+1} = x_n - \frac{f(x_n) - \alpha f(x_n) e^{-\alpha x}}{f'(x_n) - f'(x_n) e^{-\alpha x} + \alpha e^{-\alpha x} f(x_n)} \tag{20}$$

Algoritmo 9 *Sea* $g(x) = f(x) e^{\alpha x}$, *entonces el método de Newton, toma la forma*

$$x_{n+1} = x_n - \frac{f(x_n) - \alpha f(x_n) e^{\alpha x}}{f'(x_n) - f'(x_n) e^{\alpha x} - \alpha e^{\alpha x} f(x_n)} \tag{21}$$

Ahora consideremos en nuestro estudio, algunas variantes de $g(x)$ en función de la derivada de $f(x)$, las cuales se sistematizan en las fórmulas siguientes

Sea $g(x_{nn}) = \alpha f'(x_{nn})$, $\alpha \in \mathbb{R}$, y sustituyendo en el esquema iterativo anterior, obtenemos

$$x_{n+1} = x_n - \frac{f(x_n) - \alpha f'(x_n)}{f'(x_n) - \alpha f''(x_n)} \tag{22}$$

Notemos que para $\alpha = 0$, la variante obtenida se reduce al método de Newton en su forma estándar.

Consideremos la función auxiliar $g(x) = f''(x)$, entonces la nueva variante obtenida es

$$x_{n+1} = x_n - \frac{f(x_n) - f''(x_n)}{f'(x_n) - f'''(x_n)} \tag{23}$$

Generalizando el método, se obtiene que para $g(x) = f^{(n)}(x)$

$$x_{n+1} = x_n - \frac{f(x_n) - f^{(n)}(x_n)}{f'(x_n) - f^{(n+1)}(x_n)} \tag{24}$$

Ahora consideremos algunas variantes de las recíprocas de las derivadas, es decir,

Sea $g(x) = \dfrac{1}{f'(x)}$, entonces la nueva variante del método de Newton es

$$x_{n+1} = x_n - \frac{[f'(x_n)]^2 f(x_n) - f'(x_n)}{[f'(x_n)]^3 + f''(x_n)} \tag{25}$$

Si $g(x) = \dfrac{1}{f''(x)}$, entonces la variante obtenida es

$$x_{n+1} = x_n - \frac{f''(x_n)\left[f(x_n) f''(x_n) - 1\right]}{f'(x_n)\left[f''(x_n)\right]^2 + f'''(x_n)} \tag{26}$$

De manera más general, sea $g(x) = \dfrac{1}{f^{(n)}(x)}$, entonces la variante generalizada del método de Newton es

$$x_{n+1} = x_n - \frac{f^{(n)}(x_n)\left[f(x_n) f^{(n)}(x_n) - 1\right]}{f'(x_n)\left[f^{(n)}(x_n)\right]^2 + f^{(n+1)}(x_n)} \tag{27}$$

Sea $g(x) = \dfrac{f(x)}{f'(x)}$, entonces

$$x_{n+1} = x_n - f(x_n) f'(x_n) \frac{f'(x_n) - 1}{\left[f'(x_n)\right]^3 + \left[f'(x_n)\right]^2 - f(x_n) f''(x_n)} \tag{28}$$

Por otro lado, si $g(x) = \dfrac{f''(x)}{f'(x)}$, luego la variante obtenida es

$$x_{n+1} = x_n - \frac{f'(x_n) f''(x_n) - f(x_n)\left[f'(x_n)\right]^2}{\left[f'(x_n)\right]^3 + f'(x_n) f'''(x_n) - \left[f''(x_n)\right]^2} \tag{29}$$

7.2.3. Caso III : Forma funcional $h(x) = f(x) g(x)$

Ahora se presentará el caso donde la función $h(x)$ tiene la forma funcional de un producto, es decir

$$h(x) = f(x) g(x)$$

donde

$$h'(x) = f'(x) g(x) + g'(x) f(x)$$

sustituyendo en el esquema anterior, obtenemos el nuevo esquema iterativo

$$x_{n+1} = x_n - \frac{f(x_n) g(x_n)}{f'(x_n) g(x_n) + g'(x_n) f(x_n)}$$

Este caso constituye un caso particular de la técnica iterativa variacional y por ende, la función exponencial jugará un papel preponderante y principal en la obtención de nuevas variantes del método de Newton. Este caso es el más interesante, como la función exponencial es igual a su n-ésima derivada, obtenemos diferentes y nuevos algoritmos para la solución de una ecuación no lineal.

Consideremos la función $g(x)$ como una función auxiliar arbitraria que puede tomar diferentes variantes funcionales, consideremos los siguientes algoritmos

Sea $g(x) = \alpha x$, entonces, la variante obtenida es

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \tag{30}$$

lo cual constituye el método de Newton en su forma estándar.

Algoritmo 10 *Sea la función auxiliar definida por $g(x) = x$ entonces*

$$x_{n+1} = x_n - \frac{(x_n) f(x_n)}{(x_n) f'(x_n) + f(x_n)} \tag{31}$$

Algoritmo 11 *Sea la función auxiliar definida por $g(x) = f(x)$ entonces*

$$x_{n+1} = x_n - \frac{f(x_n)}{2f'(x_n)} \tag{32}$$

Sea $g(x) = f^2(x)$ entonces, la nueva variante es

$$x_{n+1} = x_n - \frac{f(x_n)}{3f'(x_n)} \tag{33}$$

Generalizando el método, consideremos la función auxiliar definida por

$$g(x) = [f(x)]^p, \; p \in \mathbb{Z}^+$$

entonces, la variante generalizada es

$$x_{n+1} = x_n - \frac{f(x_n)}{(1+p) f'(x_n)}$$

Notemos que para $p = 0$, se obtiene el método de Newton en su forma estándar.

Ahora consideremos el caso donde intervienen las derivadas.

Algoritmo 12 *Sea $g(x) = f'(x)$ entonces, la variante obtenida es*

$$x_{n+1} = x_n - \frac{f(x_n) f'(x_n)}{[f'(x_n)]^2 + f''(x_n) f(x_n)} \tag{34}$$

Algoritmo 13 *Sea la función auxiliar definida por $g(x) = f''(x)$ entonces*

$$x_{n+1} = x_n - \frac{f(x_n) f''(x_n)}{f'(x_n) f''(x_n) + f'''(x_n) f(x_n)} \tag{35}$$

Ahora vamos a considerar la recíproca de las derivadas,

Algoritmo 14 *Sea la función auxiliar definida por* $g(x) = \dfrac{1}{f'(x)}$ *entonces el esquema iterativo obtenido es*

$$x_{n+1} = x_n + \frac{f(x_n) f'(x_n)}{f''(x_n) f(x_n) - [f'(x_n)]^2} \tag{36}$$

Sea $g(x) = \dfrac{f(x)}{f'(x)}$ entonces la variante obtenida es

$$x_{n+1} = x_n - \frac{f'(x_n) + f(x_n) f''(x_n) - [f''(x_n)]^2}{[f'(x_n)]^2} \tag{37}$$

Ahora vamos a considerar la familia de funciones exponenciales, de esta manera, obtenemos múltiples variantes del método de Newton. Para precisar, consideremos la función auxiliar definida por

Algoritmo 15 *Sea* $g(x) = e^{-\alpha x}$ *entonces la nueva variente del método de Newton es*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) - \alpha f(x_n)} \tag{38}$$

Notemos que si $\alpha = 0$ el esquema obtenido se reduce al método de Newton clásico.

Algoritmo 16 *Sea la función auxiliar definida por* $g(x) = e^{-\alpha f(x)}$ *entonces el esquema obtenido es*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) - \alpha f'(x_n) f(x_n)} \tag{39}$$

Notemos que si $\alpha = 0$ el esquema obtenido se reduce al método de Newton clásico.

Algoritmo 17 *Sea* $g(x) = e^{\alpha x}$ *entonces el esquema obtenido es*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) + \alpha f(x_n)} \tag{40}$$

Algoritmo 18 *Sea* $g(x) = e^{\alpha f(x)}$ *entonces el esquema obtenido es*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) + \alpha f(x_n) f'(x_n)} \tag{41}$$

Algoritmo 19 *Sea la función auxiliar definida por* $g(x) = e^{-\alpha \frac{f(x)}{f'(x)}}$ *entonces el nuevo método obtenido es*

$$x_{n+1} = x_n - \frac{f(x_n)[f'(x_n)]^2}{[f'(x_n)]^3 - \alpha [f'(x_n)]^2 f(x_n) + \alpha f''(x_n) f^2(x_n)} \tag{42}$$

Notemos que para $\alpha = 0$, nuevamente se obtiene el método clásico de Newton.

Algoritmo 20 *Si la función auxiliar está definida por* $g(x) = e^{-\alpha \frac{f''(x)}{f'(x)}}$ *entonces el nuevo esquema iterativo es*

$$x_{n+1} = x_n - \frac{f(x_n)[f'(x_n)]^2}{[f'(x_n)]^3 - \alpha f'''(x_n) f'(x_n) f(x_n) + \alpha [f''(x_n)]^2 f(x_n)} \tag{43}$$

Notemos que para $\alpha = 0$, nuevamente se obtiene el método clásico de Newton.

7.2.4. Caso IV : Forma funcional $h(x) = f(x)/g(x)$

La función $h(x)$ tiene la forma funcional de un cociente, es decir

$$h(x) = \frac{f(x)}{g(x)}$$

donde

$$h'(x) = \frac{f'(x)g(x) - g'(x)f(x)}{[g(x)]^2}$$

sustituyendo en el esquema del método de Newton, obtenemos el nuevo esquema iterativo

$$x_{n+1} = x_n - \frac{f(x_n)g(x_n)}{f'(x_n)g(x_n) - g'(x_n)f(x_n)}$$

Ahora vamos a considerar la función $g(x)$ como una función auxiliar arbitraria que puede tomar diferentes variantes:

Algoritmo 21 *Sea la función auxiliar definida por* $g(x) = x$ *entonces*

$$x_{n+1} = x_n - \frac{(x_n)f(x_n)}{(x_n)f'(x_n) - f(x_n)} \tag{44}$$

Notemos que si la función auxiliar definida por $g(x) = f(x)$ entonces, el método colapsa.

De manera más general, sea la función auxiliar definida por

$$g(x) = [f(x)]^p, \; p \in \mathbb{Z}^+$$

entonces

$$x_{n+1} = x_n - \frac{f(x_n)}{(1-p)f'(x_n)}$$

notemos que para $p = 0$, se obtiene el método clásico de Newton.

Algoritmo 22 *Sea $g(x) = f'(x)$ entonces la nueva variante del método de Newton es*

$$x_{n+1} = x_n - \frac{f(x_n) f'(x_n)}{[f'(x_n)]^2 - f''(x_n) f(x_n)} \tag{45}$$

Algoritmo 23 *Sea la función auxiliar definida por $g(x) = f''(x)$ entonces el esquema iterativo es*

$$x_{n+1} = x_n - \frac{f(x_n) f''(x_n)}{f'(x_n) f''(x_n) - f'''(x_n) f(x_n)} \tag{46}$$

Consideremos la recíproca de la derivada, es decir,

Algoritmo 24 *Si la función auxiliar está definida por $g(x) = \dfrac{1}{f'(x)}$ entonces*

$$x_{n+1} = x_n + \frac{f(x_n) f'(x_n)}{f''(x_n) f(x_n) - [f'(x_n)]^2} \tag{47}$$

Ahora, consideremos a $g(x) = \dfrac{f(x)}{f'(x)}$ entonces la nueva fórmula obtenida es

$$x_{n+1} = x_n - \frac{f'(x_n)}{f''(x_n)} \tag{48}$$

Consideremos ahora la familia de funciones exponenciales. Para precisar, consideremos las siguiente fórmulas.

Algoritmo 25 *Sea la función auxiliar definida por $g(x) = e^{-\alpha x}$ entonces el esquema iterativo toma la forma*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) + \alpha f(x_n)} \tag{49}$$

Notemos que para $\alpha = 0$, se obtiene el método clásico de Newton.

Algoritmo 26 *Sea $g(x) = e^{-\alpha f(x)}$ entonces la nueva variante del método de Newton es*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) + \alpha f(x_n) f'(x_n)}$$

Notemos que para $\alpha = 0$, se obtiene el método clásico de Newton.

Sea la función auxiliar definida por $g(x) = e^{\alpha x}$ entonces

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) - \alpha f(x_n)} \tag{50}$$

Notemos que para $\alpha = 0$, se obtiene el método clásico de Newton.

Sea la función auxiliar definida por $g(x) = e^{\alpha f(x)}$ entonces

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) - \alpha f(x_n) f'(x_n)} \tag{51}$$

Notemos que para $\alpha = 0$, se obtiene el método clásico de Newton.

Algoritmo 27 *Sea la función auxiliar definida por* $g(x) = e^{-\alpha \frac{f(x)}{f'(x)}}$ *entonces*

$$x_{n+1} = x_n - \frac{f(x_n)[f'(x_n)]^2}{[f'(x_n)]^3 - \alpha [f'(x_n)]^2 f(x_n) + \alpha f''(x_n) f^2(x_n)} \tag{52}$$

Notemos que para $\alpha = 0$, se obtiene el método clásico de Newton.

Algoritmo 28 *Sea* $g(x) = e^{-\alpha \frac{f''(x)}{f'(x)}}$ *entonces*

$$x_{n+1} = x_n - \frac{f(x_n)[f'(x_n)]^2}{[f'(x_n)]^3 - \alpha f'''(x_n) f'(x_n) f(x_n) + \alpha [f''(x_n)]^2 f(x_n)} \tag{53}$$

Notemos que para $\alpha = 0$, se obtiene el método clásico de Newton.

7.3. Resultados Numéricos

En este acápite se presentarán algunos ejemplos donde se ilustra la eficiencia de los nuevos métodos desarrollados. La comparación se realiza entre el método de Newton y algunos de los algoritmos encontrados. Se tomará como base fundamental el número de iteraciones para encontrar dicha raíz. Se tomaron como conjunto de funciones bases, aquellas funciones que fueron consideradas en diferentes artículos de investigación, las cuales poseen características de ser funciones continuas y diferenciables.

Se han programado todos los algoritmos desarrollados en el lenguaje de alto nivel C++, bajo la filosofía de Programación Orientado a Objeto (POO) y en todos los criterios de parada de los diferentes código fuente se utiliza un nivel de tolerancia de $10e^{-16}$.

Se enumeran todos los algoritmos de manera secuencial y se seleccionan aquellos que verifican la condición de ejecutarse en un número menor de iteraciones que el métodos de Newton y aquellos que logran el mismo número de iteración. Esto se realiza con el objetivo de comparar los algoritmos óptimos y los que son equivalentes (en el número de iteraciones) al método de Newton.

Se aclara que la significación de equivalencia de método en esta tesis, significa que la raíz de la ecuación no lineal converge en un número determinado de iteraciones, aunque su expresión algebraica difiera en la estructura de su fórmula.

Ejemplo 7.1 *Consideremos la ecuacion no lineal* $e^{-x} - x^3 = 0$

Los principales hallazgos respecto a esta ecuación no lineal son :

El método de Newton alcanza la raíz de dicha ecuación en 5 iteraciones, siendo la raíz real con 20 cifras decimales 0,77288295914921012475

Los algoritmos **1, 7 y 15** alcanzan la convergencia de la raíz en 4 iteraciones, superando al método de Newton.

Los algoritmos **2, 3, 5, 6, 14, 18, 19, 20, 21, 22, 23, 24, 26, 27 y 28** alcanza la raíz en igual número de iteraciones que el método de Newton, lo cual implica que estos métodos son equivalentes.

El resto de los algoritmos de este capítulo, oscila entre **6 y 9** iteraciones, lo cual significa que no son superiores al método de Newton y que su implementación matemática y computacional no es adecuada y que por tanto, dichos algoritmos no son convenientes para ser utilizado, tanto desde el punto de vista teórico como práctico.

Ejemplo 7.2 *Consideremos la ecuacion no lineal* $x - \cos x = 0$

Los principales hallazgos respecto a esta ecuación no lineal son :

El método de Newton alcanza la raíz de dicha ecuación en 4 iteraciones, siendo la raíz real con 20 cifras decimales 0,73908513321516067229

Los algoritmos **3, 5, 6, 14, 22, 24** alcanzan la convergencia de la raíz en igual número de iteraciones que el método de Newton, lo cual implica que estos métodos son equivalentes.

Para esta ecuación, no se obtuvo ningún algoritmo que superara al método de Newton en lo que respecta al número de iteraciones.

El resto de los algoritmos de este capítulo, oscila entre **6 y 7** iteraciones, lo cual significa que no son superiores al método de Newton y que su implementación matemática y computacional no es adecuada y que por tanto, dichos algoritmos no son convenientes para ser utilizado, tanto desde el punto de vista teórico como práctico.

Ejemplo 7.3 *Consideremos la ecuacion no lineal* $e^{-x} + 2\ln x = 0$

Los principales hallazgos respecto a esta ecuación no lineal son :

El método de Newton alcanza la raíz de dicha ecuación en 5 iteraciones, siendo la raíz real con 20 cifras decimales 0,79851808532225987403

Los algoritmos **4, 17 y 25** alcanzan la convergencia de la raíz en 4 iteraciones, superando al método de Newton.

Los algoritmos **3, 6, 8, 10, 14, 17, 22 y 24** alcanza la raíz en igual número de iteraciones que el método de Newton, lo cual implica que estos métodos son equivalentes.

El resto de los algoritmos de este capítulo, oscila entre **6 y 7** iteraciones, lo cual significa que no son superiores al método de Newton y que su implementación matemática y computacional no es adecuada y que por tanto, dichos algoritmos no son convenientes para ser utilizado, tanto desde el punto de vista teórico como práctico.

Ejemplo 7.4 *Consideremos la ecuacion no lineal* $x^3 + 4x^2 - 10 = 0$

Los principales hallazgos respecto a esta ecuación no lineal son :

El método de Newton alcanza la raíz de dicha ecuación en 5 iteraciones, siendo la raíz real con 20 cifras decimales 1,36523001341409688791

El algoritmos **23** es el único que alcanza la convergencia de la raíz en 4 iteraciones, superando al método de Newton.

Los algoritmos **3, 6, 7, 14, 15, 19, 21, 22, 24 y 27** alcanza la raíz en igual número de iteraciones que el método de Newton, lo cual implica que estos métodos son equivalentes.

El resto de los algoritmos de este capítulo, oscila entre **6 y 9** iteraciones, lo cual significa que no son superiores al método de Newton y que su implementación matemática y computacional no es adecuada y que por tanto, dichos algoritmos no son convenientes para ser utilizado, tanto desde el punto de vista teórico como práctico.

Ejemplo 7.5 *Consideremos la ecuacion no lineal* $e^x - x^3 = 0$

Los principales hallazgos respecto a esta ecuación no lineal son :

El método de Newton alcanza la raíz de dicha ecuación en 4 iteraciones, siendo la raíz real con 20 cifras decimales 1,85718386020783521317

El algoritmos **21** es el único que alcanza la convergencia de la raíz en 3 iteraciones, superando al método de Newton.

Los algoritmos **3, 4, 5, 6, 7, 8, 9, 10, 12, 13, 14, 15, 19, 22, 23, 24, 25 y 27** alcanza la raíz en igual número de iteraciones que el método de Newton, lo cual implica que estos métodos son equivalentes.

El resto de los algoritmos de este capítulo, oscila entre **6 y 9** iteraciones, lo cual significa que no son superiores al método de Newton y que su implementación matemática y computacional no es adecuada y que por tanto, dichos algoritmos no son convenientes para ser utilizado, tanto desde el punto de vista teórico como práctico.

7.4. Conclusiones

Las conclusiones referentes al primer capítulo se pueden sistematizar en las siguientes aseveraciones :

- Generación de **4** nuevos esquemas iterativos que representan **28** nuevas variantes del método de Newton.

- Existencia de **8** nuevos algoritmos variantes del método de Newton que superan al método clásico en el número de iteraciones necesarias para la obtención de una raíz simple.

- El algoritmo **3** se presenta en cada una de las funciones bases analizadas, superando en algunos casos o siendo equivalente al método de Newton en la cantidad de iteraciones para lograr la raíz real.

- Todos los algoritmos obtenidos por estas variantes del método de Newton son convergente, oscilando todos ellos entre **3 y 9** iteraciones como máximo.

- De acuerdo a los resultados numéricos obtenidos, existen una gran diversidad de métodos iterativos equivalentes al método de Newton.

- El caso referido al producto funcional, es decir, $h(x) = f(x)\, g(x)$ constituye la variante más rica y variada con relación a los nuevos métodos iterativos, además, que constituye un caso particular de la técnica de variación iterativa. Además el esquema iterativo permite utilizar la familia de las funciones exponenciales, permitiendo obtener novedosos y nuevos métodos iterativos. Estos métodos serán ampliados en el capítulo III.

8. CAPÍTULO III : POLINOMIOS DE ADOMIAN

Considérese la ecuación no lineal de la forma $f(x) = 0$, $\alpha \in \mathbb{R}$ un cero simple de f. Se supone que la ecuación no lineal puede ser transformada de la siguiente manera

$$h = c + N(h)$$

donde c es una constante y N es una función no lineal. La técnica de descomposición de Adomian consiste en calcular la solución de h mediante una serie convergente de la forma

$$h = \sum_{n=0}^{\infty} h_n$$

y la función no lineal N se descompone como

$$N(h) = \sum_{n=0}^{\infty} A_n$$

donde los A_n son funciones llamadas polinomios de Adomian, que dependen de $h_0, h_1, h_2, \cdots, h_n$ y están dados por:

$$A_n = \frac{1}{n!}\frac{d^n}{d\lambda^n}\left[N\left(\sum_{i=0}^{\infty}\lambda^i h_i\right)\right]_{\lambda=0} \quad , \quad n = 0, 1, 2, 3, \cdots \tag{54}$$

Los primeros tres polinomios de Adomian son

$$\begin{aligned}
A_0 &= N\left(\sum_{i=0}^{\infty}\lambda^i h_i\right)_{\lambda=0} = N(h_0) \\
A_1 &= \frac{d}{d\lambda}\left[N\left(\sum_{i=0}^{\infty}\lambda^i h_i\right)\right]_{\lambda=0} \\
A_1 &= N'\left(\sum_{i=0}^{\infty}\lambda^i h_i\right)\left(\sum_{i=0}^{\infty} i\lambda^{i-1} h_i\right) = h_1 N'(h_0)
\end{aligned}$$

$$\begin{aligned}
A_2 &= \frac{1}{2}\frac{d^2}{d\lambda^2}\left[N\left(\sum_{i=0}^{\infty}\lambda^i h_i\right)\right]_{\lambda=0} \\
&= \frac{1}{2}\left[N''\left(\sum_{i=0}^{\infty}\lambda^i h_i\right)\left(\sum_{i=0}^{\infty} i\lambda^{i-1} h_i\right)^2 + N'\left(\sum_{i=0}^{\infty}\lambda^i h_i\right)\left(\sum_{i=0}^{\infty} i(i-1)\lambda^{i-2} h_i\right)\right]_{\lambda=0} \\
&= \frac{1}{2}\left[h_1^2 N''(h_0) + 2h_2 N'(h_0)\right] \\
&= \frac{1}{2}h_1^2 N''(h_0) + h_2 N'(h_0)
\end{aligned}$$

de esta manera, otros polinomios de Adomian pueden ser generados de manera similar.

Como

$$h = c + N(h)$$

se tiene

$$\sum_{n=0}^{\infty} h_n = c + \sum_{n=0}^{\infty} A_n$$

Por ser la serie h convergente, se tiene la siguiente importante relación

$$h_0 = c \quad y \quad h_{n+1} = A_n \ \ para\ n \geq 0$$

8.1. Construcción del Método de Newton por el método de Adomian

Sea $I \sqsubseteq \mathbb{R}$ un intervalo, $f : I \to \mathbb{R}$ una función de clase C^3 en I, $f(x) = 0$ una ecuación no lineal y α una raíz de f. Aplicando el Teorema de Taylor a $f(x-h)$ alrededor de x, se tiene

$$f(x-h) = f(x) + f'(x)(x-h-x) + \frac{f''(x)}{2!}(x-h-x)^2 + O\left(h^3\right)$$

por tanto

$$f(x-h) = f(x) - hf'(x) + h^2\frac{f''(x)}{2!} + O\left(h^3\right)$$

para una h suficientemente pequeño, se tiene

$$f(x-h) = 0 \approx f(x) - hf'(x) + h^2\frac{f''(x)}{2!}$$

de donde

$$h = \frac{f(x)}{f'(x)} + \frac{h^2}{2}\frac{f''(x)}{f'(x)}$$

haciendo

$$h = c + N(h)$$

donde

$$c = \frac{f(x)}{f'(x)} \quad y \quad N(h) = \frac{h^2}{2}\frac{f''(x)}{f'(x)}$$

para $n = 0$, obtenemos de $h = \sum_{n=0}^{\infty} h_n$ que

$$h \approx h_0 = c$$

como $f(x-h)=0$, se tiene

$$\alpha = x - h \approx x - h_0 = x - c = x - \frac{f(x)}{f'(x)}$$

finalmente, para x_{n0} suficientemente de α, la iteración viene dada por

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad , \quad n > 0 \tag{55}$$

Ahora mostraremos dos métodos iterativos de importancia, estos son : El método de Abbasbandy y el método de Chun, ambos métodos son desarrollados por la técnica de descomposición de Adomian, donde por analogía con el capítulo anterior, buscaremos nuevas funciones auxiliares que nos permitan obtener nuevos esquemas y métodos iterativos.

8.2. Método iterativo ABBS

El método iterativo de Abbs es introducido por Abbasbandy [1] y el mismo está basado en el método de descomposición de Adomian. Este método presenta una convergencia cúbica. La eficiencia computacional del método es $3^{1/3} \approx 1{,}44225$

Construcción del Método

Sea $I \subset \mathbb{R}$ un intervalo, $f : I \to \mathbb{R}$ una función de clase C^3 en I y $\alpha \in I$ una solución de la ecuación no lineal $f(x) = 0$. Aplicando el teorema de Taylor a $f(x-h)$ alrededor de x, obtenemos

$$f(x-h) = f(x) + f'(x)(x-h-x) + \frac{f''(x)}{2!}(x-h-x)^2 + O\left(h^3\right)$$

para un h suficientemente pequeño, se tiene

$$\begin{aligned} f(x-h) &= 0 \approx f(x) - hf'(x) + \frac{h^2}{2} f''(x) \\ h &= \frac{f(x)}{f'(x)} + \frac{h^2}{2}\frac{f''(x)}{f'(x)} \\ h &= h_0 + h_1 \end{aligned}$$

tomando

$$c = h_0 = \frac{f(x)}{f'(x)} \quad y \quad h_1 = N(h) = \frac{h^2}{2}\frac{f''(x)}{f'(x)}$$

resulta que

$$h = c + N(h)$$

aplicando el método de descomposición de Adomian a h, para $n=2$ se tiene que

$$h \approx h_0 + h_1 + h_2$$

entonces

$$h_0 = c \quad , \quad h_1 = A_0 \quad , \quad h_2 = A_1$$

donde

$$\begin{aligned} h_0 &= \frac{f(x)}{f'(x)} \\ h_1 &= N(h_0) = \frac{h_0^2}{2}\frac{f''(x)}{f'(x)} = \frac{f^2(x)\,f''(x)}{2\left[f'(x)\right]^3} \\ h_2 &= h_1 N'(h_0) = h_1 h_0 \frac{f''(x)}{f'(x)} = \frac{f^3(x)\left[f''(x)\right]^2}{2\left[f'(x)\right]^5} \end{aligned}$$

se tiene que

$$\alpha = x - h \approx x - \frac{f(x)}{f'(x)} - \frac{f^2(x)\,f''(x)}{2\left[f'(x)\right]^3} - \frac{f^3(x)\left[f''(x)\right]^2}{2\left[f'(x)\right]^5}$$

luego para un x_0 suficientemente cerca de α, el método iterativo de Abbasbandy está dado por

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n)\,f''(x_n)}{2\left[f'(x_n)\right]^3} - \frac{f^3(x_n)\left[f''(x_n)\right]^2}{2\left[f'(x_n)\right]^5} \quad , \quad n \geq 0 \tag{56}$$

8.3. Método iterativo de Chun

El método iterativo de Chun es introducido por Changbun Chun en [10], y el mismo está basado en el método de descomposición de Adomian. Este método presenta una convergencia de orden cuatro. Además, la eficiencia computacional del método es: $4^{1/4} \approx 1{,}41421$.

El esquema iterativo del método está dado por

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)} \\ x_{n+1} &= y_n - 2\frac{f(y_n)}{f'(y_n)} + \frac{f(y_n)\,f'(y_n)}{\left[f'(x_n)\right]^2} \quad , \quad n \geq 0 \end{aligned}$$

Construcción del método

Sea $I \subseteq \mathbb{R}$ un intervalo, $f : I \to \mathbb{R}^3$ una función de clase C^2 en I y $\alpha \in I$ una solución de la ecuación no lineal $f(x) = 0$. Aplicando el Teorema de Taylor a $f(x-h)$ alrededor de x, se tiene,

$$f(x-h) = f(x) + f'(x)(x-h-x) + O\left(h^2\right)$$

llamando $g(h) = O(h^2)$, tenemos

$$\begin{aligned} g(h) &= hf'(x) - f(x) + f(x-h) \\ g'(h) &= f'(x) - f'(x-h) \end{aligned}$$

para un h tal que $f(x-h) \approx 0$, obtenemos

$$h = \frac{g(h)}{f'(x)} + \frac{f(x)}{f'(x)}$$

y tomando

$$\begin{aligned} c &= \frac{f(x)}{f'(x)} \\ N &= \frac{g(h)}{f'(x)} \end{aligned}$$

tenemos que

$$h = c + N(h)$$

Aplicando el método de descomposición de Adomian a h, para $n = 2$, se tiene que,

$$h \approx h_0 + h_1 + h_2$$

de aqui, se obtiene que, $h_0 = c$, $h_1 = A_1$ y $h_2 = A_2$, donde

$$\begin{aligned} h_0 &= \frac{f(x)}{f'(x)} \\ h_1 &= N(h_0) = \frac{g(h_0)}{f'(x)} = \frac{f\left(x - \dfrac{f(x)}{f'(x)}\right)}{f'(x)} = \frac{f(y)}{f'(x)} \\ h_2 &= h_1 N'(h_0) = h_1 \frac{g'(h_0)}{f'(x)} = \frac{f\left(x - \dfrac{f(x)}{f'(x)}\right)}{f'(x)} \left[\frac{f'(x) - f'\left(x - \dfrac{f(x)}{f'(x)}\right)}{f'(x)^2}\right] \\ h_2 &= \frac{f\left(x - \dfrac{f(x)}{f'(x)}\right)}{[f'(x)]^2}\left[f'(x) - f'\left(x - \frac{f(x)}{f'(x)}\right)\right] \\ h_2 &= \frac{f(y)}{[f'(x)]^2}\left[f'(x) - f'(y)\right] \end{aligned}$$

como $f(x-h) \approx 0$, se tiene

$$\alpha = x - h \approx x - \frac{f(x)}{f'(x)} - 2\frac{f(y)}{f'(x)} + \frac{f(y)\, f'(y)}{[f'(x)]^2}$$

para x_0 suficientemente cerca de α, el método iterativo de Chun está dado por

$$
\begin{aligned}
y_n &= x_n - \frac{f(x_n)}{f'(x_n)} \\
x_{n+1} &= y_n - 2\frac{f(y_n)}{f'(x_n)} + \frac{f(y_n) f'(y_n)}{[f'(x_n)]^2} \quad , \quad n \geq 0
\end{aligned}
\tag{57}
$$

El objetivo central de este capítulo es considerar diferentes variantes de la función $N(h)$ y mostrar que podemos obtener diferentes esquemas iterativos y a partir de aqui, nuevos métodos iterativos, todos ellos basado en la técnica de descomposción de Adomian.

8.4. Variantes de Métodos Iterativos

Ahora vamos a considerar diferentes funciones auxiliares con el objetivo de obtener nuevos métodos iterativos.

Desarrollemos la función $f(x-h)$ en una serie de Taylor de orden 3, alrededor del punto x, entonces

$$
\begin{aligned}
f(x-h) &= f(x) + (x-h-x) f'(x) + (x-h-x)^2 \frac{f''(x)}{2} + O\left(h^3\right) \\
f(x-h) &= f(x) - hf'(x) + \frac{h^2}{2} f''(x)
\end{aligned}
$$

para un h suficientemente pequeño, se tiene $f(x-h) = 0$, luego

$$
\begin{aligned}
0 &\approx f(x) - hf'(x) + \frac{h^2}{2} f''(x) \\
h &= \frac{f(x)}{f'(x)} + \frac{h^2}{2} \frac{f''(x)}{f'(x)} \\
h &= h_0 + h_1
\end{aligned}
$$

tomando

$$
c = h_0 = \frac{f(x)}{f'(x)} \quad y \quad h_1 = N(h) = \frac{h^2}{2} \frac{f''(x)}{f'(x)}
$$

resulta que

$$
h = c + N(h)
$$

aplicando el método de descomposición de Adomian a h, para $n=4$ se tiene que

$$\begin{aligned} h &= \sum_{n=0}^{4} h_n \\ &= h_0 + h_1 + h_2 + h_3 + h_4 \end{aligned}$$

entonces

$$h_0 = c \quad , \quad h_1 = A_0 \quad , \quad h_2 = A_1 \quad , \quad h_3 = A_2 \quad , \quad h_4 = A_3$$

donde

$$\begin{aligned} h_0 &= \frac{f(x)}{f'(x)} = c \\ h_1 &= N(h_0) = \frac{h_0^2}{2}\frac{f''(x)}{f'(x)} = \frac{f^2(x) f''(x)}{2\left[f'(x)\right]^3} = A_0 \\ h_2 &= h_1 N'(h_0) = h_1 h_0 \frac{f''(x)}{f'(x)} = \frac{f^3(x)\left[f''(x)\right]^2}{2\left[f'(x)\right]^5} = A_1 \\ h_3 &= \frac{1}{2} h_1^2 N''(h_0) + h_2 N(h_0) = \frac{3}{8}\frac{f^5(x)}{\left[(f'(x)\right]^8}\left[f''(x)\right]^3 = A_2 \\ h_4 &= \frac{1}{6}\left[h_1^3 N'''(h_0) + 5 h_1 h_2 N''(h_0) + h_3 N'(h_0)\right] = \frac{1}{32}\frac{f^5(x_n)}{\left[f'(x_n)\right]^9}\left[f''(x_n)\right]^4 = A_3 \end{aligned}$$

se tiene que

$$\alpha = x - h \approx x - h_0 - h_1 - h_2 - h_3 - h_4$$

entonces, el método de Abbasbandy, toma la forma para los cuatros primeros polinomios de Adomian, en la forma del algoritmo

Algoritmo 29 *(Algoritmo AB2)*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n) f''(x_n)}{2\left[f'(x_n)\right]^3} - \frac{f^3(x_n)\left[f''(x_n)\right]^2}{2\left[f'(x_n)\right]^5} - \frac{3}{8}\frac{f^5(x_n)}{\left[(f'(x_n)\right]^8}(f''(x_n))^3$$

y para los primeros cincos polinomios, se tiene

Algoritmo 30 *(Algoritmo AB3)*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n) f''(x_n)}{2[f'(x_n)]^3} - \frac{f^3(x_n)[f''(x_n)]^2}{2[f'(x_n)]^5} - \frac{3}{8}\frac{f^5(x_n)}{[(f'(x_n)]^8}(f''(x_n))^3 - \frac{1}{32}\frac{f^5(x_n)}{(f'(x_n))^9}(f''(x_n))^4$$

Ahora vamos a considerar nuevamente el desarrollo de la serie de Taylor de orden 2, alrededor de x, es decir,

$$f(x-h) = f(x) + (x-h-x) f'(x) + O(h^2)$$

$$f(x-h) = f(x) - hf'(x) + O(h^2)$$

para un h suficientemente pequeño, se tiene $f(x-h) = 0$, luego

$$g(h) = O(h^2)$$

$$g(h) = hf'(x) - f(x)$$

$$h = \frac{f}{f'} + \frac{g(h)}{f'}$$

$$h = h_0 + h_1$$

entonces, hagamos $g(h) = f(y)$, donde $y = x - h$, al realizar los cálculos sobre los polinomios de Adomian, tenemos que

$$h_0 = \frac{f}{f'} = c$$

$$h_1 = \frac{g(h_0)}{f'(x)} = N(h_0) = \frac{f(x-h)}{f'(x)} = \frac{f(y)}{f'(x)}$$

$$N'(h_0) = \frac{-f'(x-h)}{f'(x)}$$

$$h_2 = h_1 N'(h_0) = \left(\frac{f(x-h)}{f'(x)}\right)\left(\frac{-f'(x-h)}{f'(x)}\right) = -\frac{f(y)f'(y)}{[f'(x)]^2}$$

luego, obtenemos un nuevo método iterativo, definido por

Algoritmo 31

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n)} + \dfrac{f(y_n)\, f'(y_n)}{\left[f'(x_n)\right]^2} \end{array}\right. \tag{58}$$

Considerando nuevamente el desarrollo de la serie de Taylor de orden 2 y haciendo $g(h) = \dfrac{f' g(h)}{f'}$, al realizar los cálculos sobre los polinomios de Adomian, tenemos que

$$\begin{aligned} h_0 &= \frac{f}{f'} = c \\ h_1 &= \frac{f' g(h_0)}{(f')^2} = N(h_0) = \frac{f' g(h_0)}{(f')^2} = \frac{f\left(x - \dfrac{f}{f'}\right)}{f'} = \frac{f(y)}{f'(x)} \\ N'(h_0) &= \frac{g'(h_0)}{(f')^2} = \frac{f'(x) - f'(x - h_0)}{(f')^2} = \frac{f'(x) - f'(y_n)}{\left[f'(x)\right]^2} \\ h_2 &= h_1 N'(h_0) = \left(\frac{f(y)}{f'(x)}\right)\left(\frac{g'(h_0)}{f'(x)}\right) = \left(\frac{f(y)}{f'(x)}\right)\left(\frac{f'(x) - f'(y)}{\left[f'(x)\right]^2}\right) \end{aligned}$$

luego, obtenemos un nuevo método iterativo, definido por

Algoritmo 32

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n)} - \dfrac{f(y_n)}{f'(x_n)}\left(\dfrac{f'(x_n) - f'(y_n)}{\left(f'(x_n)\right)^2}\right) \end{array}\right. \tag{59}$$

Considerando nuevamente el desarrollo de la serie de Taylor de orden 2 y haciendo la función auxiliar

$$\begin{aligned} g(h) &= h f(x - h) \\ g'(h) &= f(x - h) - f'(x - h)\, h \end{aligned}$$

y sabiendo que

$$
\begin{aligned}
h &= \frac{f}{f'} + \frac{g(h)}{f'} \\
h &= h_0 + h_1
\end{aligned}
$$

entonces los polinomios de Adomian, toman las formas

$$
\begin{aligned}
h_0 &= \frac{f}{f'} = c \\
h_1 &= \frac{g(h_0)}{f'} = N(h_0) = \frac{hf(x-h)}{f'} = \frac{f(x)f(y)}{[f'(x)]^2} \\
N'(h_0) &= \frac{g'(h_0)}{f'} = \frac{f(y) - hf'(y)}{f'} \\
h_2 &= h_1 N'(h_0) = \left(\frac{f(x)f(y)}{[f'(x)]^2}\right)\left(\frac{f(y)-hf'(y)}{f'}\right) = \frac{f(x)}{[f'(x)]^3}[f(y)]^2\left(1 - \frac{f(x)}{f'(x)}\right)
\end{aligned}
$$

luego, obtenemos un nuevo método iterativo, definido por

Algoritmo 33

$$
\left\{
\begin{array}{c}
y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\
\\
x_{n+1} = y_n - \dfrac{f(x_n)f(y_n)}{[f'(x_n)]^2} - \dfrac{f(x_n)[f(y_n)]^2}{[f'(x_n)]^3}\left(1 - \dfrac{f(x_n)}{f'(x_n)}\right)
\end{array}
\right.
\tag{60}
$$

Considerando nuevamente el desarrollo de la serie de Taylor de orden 2 y haciendo la función auxiliar

$$
\begin{aligned}
g(h) &= h^2 f(y) \\
g'(h) &= 2hf(y) - h^2 f'(y)
\end{aligned}
$$

donde $y = x - h$, y sabiendo que

$$
\begin{aligned}
h &= \frac{f}{f'} + \frac{g(h)}{f'} \\
h &= h_0 + h_1
\end{aligned}
$$

luego los polinomios de Adomian son

$$\begin{aligned}
h_0 &= \frac{f}{f'} = c \\
h_1 &= \frac{g\left(h_0\right)}{f'\left(x\right)} = N\left(h_0\right) = \frac{h^2 f\left(y\right)}{f'\left(x\right)} = \frac{f^2\left(x\right) f\left(y\right)}{\left[f'\left(x\right)\right]^3} \\
N'\left(h_0\right) &= \frac{g'\left(h_0\right)}{f'} = \frac{2hf\left(y\right) - h^2 f'\left(y\right)}{f'\left(x\right)} = \frac{2f\left(x\right) f\left(y\right)}{\left[f'\left(x_n\right)\right]^2} - \frac{f^2\left(x\right) f'\left(y\right)}{\left[f'\left(x_n\right)\right]^3} \\
h_2 &= h_1 N'\left(h_0\right) = \left(\frac{f^2\left(x\right) f\left(y\right)}{\left[f'\left(x\right)\right]^3}\right)\left(\frac{2f\left(x\right) f\left(y\right)}{\left[f'\left(x_n\right)\right]^2} - \frac{f^2\left(x\right) f'\left(y\right)}{\left[f'\left(x_n\right)\right]^3}\right) = \frac{2f^3\left(x\right) f^2\left(y\right)}{\left[f'\left(x\right)\right]^5} - \frac{2f^4\left(x\right) f\left(y\right) f'\left(y\right)}{\left[f'\left(x\right)\right]^6}
\end{aligned}$$

de donde, obtenemos un nuevo método iterativo, definido por

Algoritmo 34

$$\begin{cases}
y_n = x_n - \dfrac{f\left(x_n\right)}{f'\left(x_n\right)} \\
\\
x_{n+1} = y_n - \dfrac{f^2\left(x_n\right) f\left(y_n\right)}{\left[f'\left(x_n\right)\right]^3} - \dfrac{2f^3\left(x_n\right) f^2\left(y_n\right)}{\left[f'\left(x_n\right)\right]^5} + \dfrac{2f^4\left(x_n\right) f\left(y_n\right) f'\left(y_n\right)}{\left[f'\left(x_n\right)\right]^6}
\end{cases} \tag{61}$$

Considerando nuevamente el desarrollo de la serie de Taylor de orden 2 y haciendo la función auxiliar

$$\begin{aligned}
g\left(h\right) &= \left[f'\left(x\right)\right]^2 f\left(y\right) \\
g'\left(h\right) &= -\left[f'\left(x\right)\right]^2 f'\left(y\right)
\end{aligned}$$

donde $y = x - h$, y sabiendo que

$$\begin{aligned}
h &= \frac{f}{f'} + \frac{g\left(h\right)}{f'} \\
h &= h_0 + h_1
\end{aligned}$$

entonces, los polinomios de Adomian son

$$
\begin{aligned}
h_0 &= \frac{f}{f'} = c \\
h_1 &= \frac{g\left(h_0\right)}{f'\left(x\right)} = N\left(h_0\right) = \frac{f\left(y\right)}{f'\left(x\right)} \\
N'\left(h_0\right) &= -\frac{f'\left(y\right)}{f'\left(x\right)} \\
h_2 &= h_1 N'\left(h_0\right) = \left(\frac{f\left(y\right)}{f'\left(x\right)}\right)\left(-\frac{f'\left(y\right)}{f'\left(x\right)}\right) = -\frac{f\left(y\right) f'\left(y\right)}{\left[f'\left(x\right)\right]^2}
\end{aligned}
$$

de manera que el nuevo método iterativo es

Algoritmo 35

$$
\begin{cases}
y_n = x_n - \dfrac{f\left(x_n\right)}{f'\left(x_n\right)} \\
x_{n+1} = y_n - \dfrac{f\left(y_n\right)}{f'\left(x_n\right)} + \dfrac{f\left(y_n\right) f'\left(y_n\right)}{\left[f'\left(x_n\right)\right]^2}
\end{cases}
\tag{62}
$$

Considerando nuevamente el desarrollo de la serie de Taylor de orden 2 y considerando la función auxiliar

$$
\begin{aligned}
g\left(h\right) &= \frac{f\left(x-h\right)}{f'} \\
g'\left(h\right) &= \frac{-f'\left(y\right)}{f'}
\end{aligned}
$$

donde $y = x - h$, y sabiendo que

$$
\begin{aligned}
h &= \frac{f}{f'} + \frac{g\left(h\right)}{f'} \\
h &= h_0 + h_1
\end{aligned}
$$

entonces

$$
\begin{aligned}
h_0 &= \frac{f}{f'} = c \\
h_1 &= \frac{g\left(h_0\right)}{f'} = N\left(h_0\right) = \frac{f\left(y\right)}{f'\left(x\right)}
\end{aligned}
$$

$$
\begin{aligned}
N'(h_0) &= \frac{-f'(y)}{f'(x)} \\
h_2 &= h_1 N'(h_0) = \left(\frac{f(y)}{f'(x)}\right)\left(\frac{-f'(y)}{f'(x)}\right) = -\frac{f(y)\, f'(y)}{[f'(x)]^2}
\end{aligned}
$$

de donde, se obtiene el método iterativo

Algoritmo 36

$$
\begin{cases}
y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\
\\
x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n)} + \dfrac{f(y_n)\, f'(y_n)}{[f'(x_n)]^2}
\end{cases}
\tag{63}
$$

Considerando nuevamente el desarrollo de la serie de Taylor de orden 2 y considerando una variante de la función auxiliar, definida por

$$
\begin{aligned}
g(h) &= f(x)\, f(y) \\
g'(h) &= -f(x)\, f'(y)
\end{aligned}
$$

donde $y = x - h$, y sabiendo que

$$
\begin{aligned}
h &= \frac{f}{f'} + \frac{g(h)}{f'} \\
h &= h_0 + h_1
\end{aligned}
$$

entonces, los polinomios de Adomian son

$$
\begin{aligned}
h_0 &= \frac{f}{f'} = c \\
h_1 &= \frac{g(h)}{f'} = \frac{f(x)\, f(y)}{f'(x)} = N(h_0) \\
N'(h_0) &= \frac{g'(h)}{f'(x)} = -\frac{f(x)\, f'(y)}{f'(x)} \\
h_2 &= h_1 N'(h_0) = \left(\frac{f(x)\, f(y)}{f'(x)}\right)\left(-\frac{f(x)\, f'(y)}{f'(x)}\right) = -\frac{f^2(x)\, f(y)\, f'(y)}{[f'(x)]^2}
\end{aligned}
$$

luego, el método iterativo está definido por

Algoritmo 37

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n)\,f(y_n)}{f'(x_n)} + \dfrac{f^2(x_n)\,f(y_n)\,f'(y_n)}{\left[f'(x_n)\right]^2} \end{cases} \tag{64}$$

Considerando nuevamente el desarrollo de la serie de Taylor de orden 2 y haciendo la función auxiliar

$$\begin{aligned} g(h) &= f^2(y) \\ \\ g'(h) &= -2f(y)\,f'(y) \end{aligned}$$

donde $y = x - h$, y sabiendo que

$$\begin{aligned} h &= \frac{f}{f'} + \frac{g(h)}{f'} \\ \\ h &= h_0 + h_1 \end{aligned}$$

entonces, los polinomios de Adomian están definido por

$$\begin{aligned} h_0 &= \frac{f}{f'} = c \\ \\ h_1 &= \frac{g(h)}{f'} = \frac{f^2(y)}{f'(x)} = N(h_0) \\ \\ N'(h_0) &= \frac{g'(h)}{f'(x)} = -\frac{2f(y)\,f'(y)}{f'(x)} \\ \\ h_2 &= h_1 N'(h_0) = \left(\frac{f^2(y)}{f'(x)}\right)\left(-\frac{2f(y)\,f'(y)}{f'(x)}\right) = -\frac{2f^3(y)\,f'(y)}{\left[f'(x)\right]^2} \end{aligned}$$

luego, el nuevo método iterativo es

Algoritmo 38

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^2(y_n)}{f'(x_n)} + \dfrac{2f^3(y_n)\,f'(y_n)}{\left[f'(x_n)\right]^2} \end{cases} \tag{65}$$

Consideremos una nueva variante definido por $N\left(h\right)=\dfrac{g\left(h\right)}{f'\left(x\right)}$ entonces $N'\left(h\right)=\dfrac{g'\left(h\right)}{f'\left(x\right)}$, donde $g\left(h\right)=f\left(x-h\right)-f\left(x\right)+hf'\left(x\right)$, en consecuencia

$$N'\left(h\right)=\frac{g'\left(h\right)}{f'\left(x\right)}=\frac{f'\left(x\right)-f'\left(y\right)}{f'\left(x\right)}$$

donde $y=x-h$, y por tanto,

$$\begin{aligned}
h_0 &= \frac{f}{f'}=c \\
h_1 &= \frac{g\left(h\right)}{f'}=\frac{f\left(y\right)}{f'\left(x\right)}=N\left(h_0\right) \\
h_2 &= h_1N'\left(h_0\right)=\left(\frac{f\left(y\right)}{f'\left(x\right)}\right)\left(\frac{f'\left(x\right)-f'\left(y\right)}{f'\left(x\right)}\right)
\end{aligned}$$

entonces, el nuevo método iterativo es

Algoritmo 39

$$\left\{\begin{array}{c}
y_n=x_n-\dfrac{f\left(x_n\right)}{f'\left(x_n\right)} \\
\\
x_{n+1}=y_n-\dfrac{f\left(y_n\right)}{f'\left(x_n\right)}-\dfrac{f\left(y_n\right)f'\left(x_n\right)-f\left(y_n\right)f'\left(y_n\right)}{\left[f'\left(x_n\right)\right]^2}
\end{array}\right. \tag{66}$$

Consideremos ahora $N\left(h\right)=\dfrac{g\left(h\right)}{f'\left(x\right)}$ entonces $N'\left(h\right)=\dfrac{g'\left(h\right)}{f'\left(x\right)}$, donde $g\left(h\right)=\dfrac{f\left(x-h\right)-f\left(x\right)+hf'\left(x\right)}{f'\left(x\right)}$, en consecuencia

$$\begin{aligned}
g'\left(h\right) &= -\frac{f'\left(y\right)}{f'\left(x\right)} \\
N'\left(h\right) &= \frac{g'\left(h\right)}{f'\left(x\right)}=\frac{-f'\left(y\right)}{\left[f'\left(x\right)\right]^2}
\end{aligned}$$

donde $y=x-h$ y por tanto,

$$\begin{aligned}
h_0 &= \frac{f}{f'}=c \\
h_1 &= \frac{g\left(h\right)}{f'}=\frac{\dfrac{f\left(x-h\right)-f\left(x\right)+hf'\left(x\right)}{f'\left(x\right)}}{f'\left(x\right)}=\frac{f\left(y\right)}{\left[f'\left(x\right)\right]^2}=N\left(h_0\right) \\
h_2 &= h_1N'\left(h_0\right)=-\left(\frac{f\left(y\right)}{\left[f'\left(x\right)\right]^2}\right)\left(\frac{f'\left(y\right)}{\left[f'\left(x\right)\right]^2}\right)=-\frac{f\left(y\right)f'\left(y\right)}{\left[f'\left(x\right)\right]^4}
\end{aligned}$$

entonces, el nuevo método iterativo es

Algoritmo 40

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{[f'(x_n)]^2} + \dfrac{f(y_n)\, f'(y_n)}{[f'(x_n)]^4} \end{cases} \tag{67}$$

Sea $N(h) = \dfrac{g(h)}{f'(x)}$ entonces $N'(h) = \dfrac{g'(h)}{f'(x)}$, donde $g(h) = f(x)\, f(y)$, en consecuencia

$$g'(h) = f'(x)\, f(y) - f(x)\, f'(y)$$

$$N'(h) = \frac{g'(h)}{f'(x)} = \frac{f'(x)\, f(y) - f(x)\, f'(y)}{f'(x)}$$

donde $y = x - h$ y por tanto,

$$\begin{aligned} h_0 &= \frac{f}{f'} = c \\ h_1 &= \frac{g(h)}{f'} = \frac{f(x)\, f(y)}{f'(x)} = N(h_0) \\ h_2 &= h_1 N'(h_0) = \left(\frac{f(x)\, f(y)}{f'(x)}\right)\left(\frac{f'(x)\, f(y) - f(x)\, f'(y)}{f'(x)}\right) \\ h_2 &= \frac{f(x)\, f'(x)\, f(y) - f^2(x)\, f(y)\, f'(y)}{[f'(x)]^2} \end{aligned}$$

entonces, obtenemos el siguiente algoritmo

Algoritmo 41

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n)\, f(y_n)}{f'(x_n)} - \dfrac{f(x_n)\, f'(x_n)\, f(y_n) - f^2(x_n)\, f(y_n)\, f'(y_n)}{[f'(x_n)]^2} \end{cases} \tag{68}$$

Considerando nuevamente el desarrollo de la serie de Taylor de orden 2 y haciendo la función auxiliar

$$\begin{aligned} g(h) &= f(y) \\ g'(h) &= -f'(y) \end{aligned}$$

y sabiendo que

$$\begin{aligned} h &= \frac{f}{f'} + \frac{g(h)}{f'} \\ h &= h_0 + h_1 \end{aligned}$$

entonces, los polinomios de Adomian

$$\begin{aligned} h_0 &= \frac{f}{f'} = c \\ h_1 &= \frac{g(h_0)}{f'} = N(h_0) = \frac{f(y)}{f'(x)} \\ N'(h_0) &= \frac{g'(h_0)}{f'} = -\frac{f'(y)}{f'(x)} \\ h_2 &= h_1 N'(h_0) = -\left(\frac{f(y)}{f'(x)}\right)\left(\frac{f'(y)}{f'(x)}\right) = -\frac{f(y)\,f'(y)}{[f'(x)]^2} \end{aligned}$$

entonces, tenemos el algoritmo

Algoritmo 42

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n)} + \dfrac{f(y_n)\,f'(y_n)}{[f'(x_n)]^2} \end{cases} \tag{69}$$

Considerando nuevamente el desarrollo de la serie de Taylor de orden 2 y considerando la función auxiliar

$$\begin{aligned} g(h) &= \frac{g(h)\,f'}{f'f'} \\ g'(h) &= \frac{g'(h)\,f'}{[f']^2} \end{aligned}$$

y sabiendo que

$$h = \frac{f}{f'} + \frac{g(h)}{f'}$$

$$h = h_0 + h_1$$

entonces los polinomios de Adomian son

$$h_0 = \frac{f}{f'} = c$$

$$h_1 = \frac{g(h_0)}{f'} = N(h_0) = \frac{g(h)\,f'}{f'f'} = \frac{f(y)\,f'(x)}{[f'(x)]^2}$$

$$N'(h_0) = \frac{g'(h)\,f'}{[f']^2}$$

$$h_2 = h_1 N'(h_0) = \left(\frac{g(h)\,f'}{[f']^2}\right)\left(\frac{g'(h)\,f'}{[f']^2}\right) = \frac{g(h)\,g'(h)\,[f'(x)]^2}{[f'(x)]^4}$$

$$h_2 = \frac{f(y)\,[f'(x) - f'(y)]}{[f'(x)]^2}$$

de donde, el método iterativo es

Algoritmo 43

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)\,f'(x_n)}{[f'(x_n)]^2} - \dfrac{f(y_n)\,[f'(x_n) - f'(y_n)]}{[f'(x_n)]^2} \end{cases} \tag{70}$$

Considerando nuevamente el desarrollo de la serie de Taylor de orden 2 y haciendo la función auxiliar

$$g(h) = \frac{g(h)\,f''}{f'f''}$$

$$g'(h) = \frac{g'(h)\,f''}{f'f''}$$

y sabiendo que

$$h = \frac{f}{f'} + \frac{g(h)}{f'}$$

$$h = h_0 + h_1$$

entonces, los polinomios de Adomian están dados por

$$h_0 = \frac{f}{f'} = c$$

$$h_1 = \frac{g(h_0)}{f'} = N(h_0) = \frac{g(h)\, f''}{f' f''} = \frac{f(y)}{f'(x)}$$

$$N'(h_0) = \frac{g'(h)\, f''}{f' f''}$$

$$h_2 = h_1 N'(h_0) = \left(\frac{g(h)\, f''}{f' f''}\right)\left(\frac{g'(h)\, f''}{f' f''}\right) = \frac{g(h)\, g'(h)\, [f''(x)]^2}{[f'(x)\, f''(x)]^2}$$

$$h_2 = \frac{f(y)\,[f'(x) - f'(y)]\,[f''(x)]^2}{[f'(x)]^2}$$

luego, el método iterativo es

Algoritmo 44

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n)} - \dfrac{f(y_n)}{[f'(x_n)]^2}\,[f'(x_n) - f'(y_n)] \end{array}\right. \tag{71}$$

Sea $N(h) = \dfrac{g(h)}{f'(x)}$ entonces $N'(h) = \dfrac{g'(h)}{f'(x)}$, donde $g(h) = \dfrac{f(y)}{h}$, en consecuencia

$$N'(h) = \frac{g'(h)}{hf'(x)}$$

$$= \frac{-f'(y)\,h - f(y)}{[hf'(x)]^2}$$

$$= \frac{-f'(y)\,h - f(y)}{f^2(x)}$$

por tanto, los polinomios de Adomian son

$$
\begin{aligned}
h_0 &= \frac{f}{f'} = c \\
h_1 &= \frac{g(h)}{f'} = \frac{f(y)}{f(x)} = N(h_0) \\
N'(h_0) &= -\frac{f'(y)\dfrac{f(x)}{f'(x)} + f(y)}{f^2(x)} \\
h_2 &= h_1 N'(h_0) = -\left(\frac{f(y)}{f(x)}\right)\left(\frac{f'(y)\dfrac{f(x)}{f'(x)} + f(y)}{f^2(x)}\right) \\
h_2 &= -\frac{f(y)\,f'(y)\,f(x) + f^2(y)\,f'(x)}{f^3(x)\,f'(x)}
\end{aligned}
$$

luego, se obtiene el siguiente algoritmo

Algoritmo 45

$$
\left\{
\begin{array}{c}
y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\
\\
x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n)} + \dfrac{f(y_n)\,f'(y_n)\,f(x_n) + f^2(y_n)\,f'(x_n)}{f^3(x_n)\,f'(x_n)}
\end{array}
\right. \tag{72}
$$

Sea $N(h) = \dfrac{g(h)}{f'(x)}$ entonces $N\prime(h) = \dfrac{g'(h)}{f'(x)}$, donde $g(h) = \dfrac{f''(x)\,g(h)}{f'(x)}$, en consecuencia

$$
\begin{aligned}
N\prime(h) &= \frac{f''(x)\,g'(h)}{f'(x)} \\
&= \frac{f''(x)\left[f'(x) - f'(y)\right]}{\left[f'(x)\right]^2}
\end{aligned}
$$

por tanto,

$$
\begin{aligned}
h_0 &= \frac{f}{f'} = c \\
h_1 &= \frac{g(h)}{f'} = \frac{f(y)}{f'(x)} = N(h_0) \\
N'(h_0) &= \frac{f''(x)\left[f'(x) - f'(y)\right]}{\left[f'(x)\right]^2} \\
h_2 &= h_1 N'(h_0) = \left(\frac{f(y)}{f'(x)}\right)\left(\frac{f''(x)\left[f'(x) - f'(y)\right]}{\left[f'(x)\right]^2}\right)
\end{aligned}
$$

entonces, el nuevo método iterativo está dado por el siguiente

$$
\left\{
\begin{array}{c}
y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\
\\
x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n)} - \dfrac{f(y_n)\, f''(x_n)}{\left[f'(x_n)\right]^3}\left[f'(x_n) - f'(y_n)\right]
\end{array}
\right. \tag{73}
$$

Considerando nuevamente el desarrollo de la serie de Taylor de orden 2 y haciendo la función auxiliar

$$
\begin{aligned}
g(h) &= f(y) \\
g'(h) &= f'(x) - f'(y)
\end{aligned}
$$

y sabiendo que

$$
\begin{aligned}
h &= \frac{f}{f'} + \frac{g(h)}{f'} \\
h &= h_0 + h_1
\end{aligned}
$$

entonces, los polinomios de Adomian están expresado por

$$
\begin{aligned}
h_0 &= \frac{f}{f'} = c \\
h_1 &= \frac{g\left(h_0\right)}{f'} = N\left(h_0\right) = \frac{f\left(y\right)}{f'\left(x\right)} \\
N'\left(h_0\right) &= \frac{g'\left(h\right)}{f'\left(x\right)} = \frac{f'\left(x\right) - f'\left(y\right)}{f'\left(x\right)} \\
h_2 &= h_1 N'\left(h_0\right) = \left(\frac{f\left(y\right)}{f'\left(x\right)}\right)\left(\frac{f'\left(x\right) - f'\left(y\right)}{f'\left(x\right)}\right) \\
h_2 &= \frac{f\left(y\right)\left[f'\left(x\right) - f'\left(y\right)\right]}{\left[f'\left(x\right)\right]^2}
\end{aligned}
$$

luego, el algoritmo obtenido es

Algoritmo 46

$$
\begin{cases}
y_n = x_n - \dfrac{f\left(x_n\right)}{f'\left(x_n\right)} \\
x_{n+1} = y_n - \dfrac{f\left(y_n\right)}{f'\left(x_n\right)} - \dfrac{f\left(y_n\right)\left[f'\left(x_n\right) - f'\left(y_n\right)\right]}{\left[f'\left(x_n\right)\right]^2}
\end{cases} \tag{74}
$$

Considerando nuevamente el desarrollo de la serie de Taylor de orden 2 y haciendo la función auxiliar

$$
\begin{aligned}
g\left(h\right) &= \frac{f\left(y\right)}{hf'\left(x\right)} = \frac{f\left(y\right)}{f\left(x\right)} \\
g'\left(h\right) &= \frac{-f'\left(y\right)f\left(x\right) - f'\left(x\right)f\left(y\right)}{f^2\left(x\right)}
\end{aligned}
$$

y sabiendo que

$$
\begin{aligned}
h &= \frac{f}{f'} + \frac{g\left(h\right)}{f'} \\
h &= h_0 + h_1
\end{aligned}
$$

entonces, los polinomios de Adomian

$$
\begin{aligned}
h_0 &= \frac{f}{f'} = c \\
h_1 &= \frac{g(h_0)}{f'} = N(h_0) = \frac{f(y)}{f(x) f'(x)} \\
N'(h_0) &= \frac{-f'(y) f(x) f'(x) - [f(x) f'(x)]' f(y)}{[f(x) f'(x)]^2} \\
&= -\frac{f(y) f'(y) f(x) + [f'(x)]^2 f(y) + f''(x) f(x) f(y)}{f^2(x) [f'(x)]^2} \\
h_2 &= h_1 N'(h_0) = -\left(\frac{f(y)}{f(x) f'(x)}\right)\left(\frac{f(y) f'(y) f(x) + [f'(x)]^2 f(y) + f''(x) f(x) f(y)}{f^2(x) [f'(x)]^2}\right) \\
h_2 &= -f^2(y) \frac{f'(y) f(x) + [f'(x)]^2 + f''(x) f(x)}{f^3(x) [f'(x)]^3}
\end{aligned}
$$

luego, el nuevo método iterativo está definido por

Algoritmo 47

$$
\begin{cases}
y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\
x_{n+1} = y_n - \dfrac{f(y_n)}{f(x_n) f'(x_n)} + f^2(y_n) \dfrac{f'(y_n) f(x_n) + [f'(x_n)]^2 + f''(x_n) f(x_n)}{f^3(x_n) [f'(x_n)]^3}
\end{cases}
\tag{75}
$$

Considerando nuevamente el desarrollo de la serie de Taylor de orden 2 y haciendo la función auxiliar

$$
\begin{aligned}
g(h) &= \frac{g(h) f(y)}{f' f(y)} \\
g'(h) &= \frac{g'(h) f(y)}{f' f(y)}
\end{aligned}
$$

y sabiendo que

$$
\begin{aligned}
h &= \frac{f}{f'} + \frac{g(h)}{f'} \\
h &= h_0 + h_1
\end{aligned}
$$

entonces, los polinomios de Adomian son

$$
\begin{aligned}
h_0 &= \frac{f}{f'} = c \\
h_1 &= \frac{g(h_0)}{f'} = N(h_0) = \frac{g(h) f(y)}{f'(x) f(y)} = \frac{f(y)}{f'(x)} \\
N'(h_0) &= \frac{g'(h) f(y)}{f'(x) f(y)} \\
h_2 &= h_1 N'(h_0) = \left(\frac{g(h) f(y)}{f'(x) f(y)} \right) \left(\frac{g'(h) f(y)}{f'(x) f(y)} \right) = \frac{g(h) g'(h)}{[f'(x)]^2} \\
h_2 &= \frac{f(y)[f'(x) - f'(y)]}{[f'(x)]^2}
\end{aligned}
$$

entonces se tiene la siguiente fórmula

$$
\begin{cases}
y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\
x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n)} - \dfrac{f(y_n)[f'(x_n) - f'(y_n)]}{[f'(x_n)]^2}
\end{cases}
\tag{76}
$$

Notemos que este algoritmo es equivalente al algoritmo **47.**

Considerando nuevamente el desarrollo de la serie de Taylor de orden 2 y haciendo la función auxiliar

$$
\begin{aligned}
g(h) &= \frac{g(h) f'(y)}{f'(x) f'(y)} \\
g'(h) &= \frac{g'(h) f'(y)}{f'(x) f'(y)}
\end{aligned}
$$

y sabiendo que

$$
h = \frac{f}{f'} + \frac{g(h)}{f'}
$$

entonces, los polinomios de Adomian son

$$\begin{aligned}
h_0 &= \frac{f}{f'} = c \\
h_1 &= \frac{g(h_0)}{f'} = N(h_0) = \frac{g(h)\,f'(y)}{f'f'(y)} = \frac{f(y)\,f'(y)}{f'f'(y)} \\
N'(h_0) &= \frac{g'(h)\,f'(y)}{f'(x)\,f'(y)} = \frac{[f'(x) - f'(y)]\,f(y)}{f'(x)\,f'(y)} \\
h_2 &= h_1 N'(h_0) = \left(\frac{f(y)\,f'(y)}{f'f'(y)}\right)\left(\frac{[f'(x) - f'(y)]\,f(y)}{f'(x)\,f'(y)}\right) \\
h_2 &= \frac{[f(y)]^2\,[f'(x) - f'(y)]}{f'(x)\,f'(y)}
\end{aligned}$$

y el algoritmo obtenido es

Algoritmo 48

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\[2ex] x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n)} - \dfrac{[f(y_n)]^2\,[f'(x_n) - f'(y_n)]}{f'(x_n)\,f'(y_n)} \end{cases} \tag{77}$$

Consideremos $g(h) = \dfrac{g(h)\,f(y)}{f'(x)\,f(y)}$, entonces por sustituciones algebraicas, obtenemos los polinomios de Adomian

$$\begin{aligned}
h_0 &= \frac{f}{f'} = c \\
h_1 &= \frac{g(h_0)\,f''}{f'f''} = N(h_0) \\
N'(h_0) &= \frac{f''}{f'f''} g'(h_0) = \frac{f''}{f'f''}\left(f'(x) - f'(x - h_0)\right) \\
h_2 &= h_1 N'(h_0) = \left(\frac{g(h_0)\,f''}{f'f''}\right)\left(\frac{f''}{f'f''}\left(f'(x) - f'(x - h_0)\right)\right) \\
&= \left(\frac{f''}{f'f''} f\left(x - \frac{f}{f'}\right)\right)\left(\frac{f''}{f'f''}\left(f'(x) - f'(x - h_0)\right)\right)
\end{aligned}$$

entonces, el nuevo método iterativo está definido por el

Algoritmo 49

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n)} - \dfrac{f(y_n)}{f'(x_n)}\left(\dfrac{f'(x_n) - f'(y_n)}{(f'(x_n))^2}\right) \end{cases} \tag{78}$$

8.5. Resultados Numéricos

En este acápite se presentarán algunos ejemplos donde se ilustra la eficiencia de los nuevos métodos desarrollados. La comparación se realiza entre el método de Newton y algunos de los algoritmos encontrados. Se tomará como base fundamental el número de iteraciones para encontrar dicha raíz. Se tomaron como conjunto de funciones bases, aquellas funciones que fueron consideradas en diferentes artículos de investigación, las cuales poseen características de ser funciones continuas y diferenciables.

Se han programado todos los algoritmos desarrollados en el lenguaje de alto nivel C++, bajo la filosofía de Programación Orientado a Objeto (POO) y en todos los criterios de parada de los diferentes códigos fuente se utiliza un nivel de tolerancia de $10e^{-16}$.

Se enumeran todos los algoritmos de manera secuencial y se seleccionan aquellos que verifican la condición de ejecutarse en un número menor de iteraciones que el método de Newton y aquellos que logran el mismo número de iteración. Esto se realiza con el objetivo de comparar los algoritmos óptimos y los que son equivalentes (en el número de iteraciones) al método de Newton.

Se aclara que la significación de equivalencia de método en esta tesis, significa que la raíz de la ecuación no lineal converge en un número determinado de iteraciones, aunque su expresión algebraica difiera en la estructura de su fórmula.

Ejemplo 8.1 *Consideremos la ecuacion no lineal* $e^{-x} - x^3 = 0$

Los principales hallazgos respecto a esta ecuación no lineal son :

El método de Newton alcanza la raíz de dicha ecuación en **5** iteraciones, siendo la raíz real con 20 cifras decimales 0,77288295914921012475

Los algoritmos **Ab2, Ab3, 9, 11, 13, 14, 16 y 18** alcanzan la convergencia de la raíz en **4** iteraciones, superando al método de Newton.

Los algoritmos **2, 4, 7 y 19** alcanza la raíz en igual número de iteraciones que el método de Newton, lo cual implica que estos métodos son equivalentes.

El resto de los algoritmos de este capítulo, oscila entre **6 y 9** iteraciones, lo cual significa que no son superiores al método de Newton y que su implementación matemática y computacional no es adecuada y que por tanto, dichos algoritmos no son convenientes para ser utilizado, tanto desde el punto de vista teórico como práctico.

Ejemplo 8.2 *Consideremos la ecuacion no lineal* $x - \cos x = 0$

Los principales hallazgos respecto a esta ecuación no lineal son :

El método de Newton alcanza la raíz de dicha ecuación en **4** iteraciones, siendo la raíz real con 20 cifras decimales 0,73908513321516067229

Los algoritmos **Ab2, Ab3, 2, 9, 13, 14, 16, 18 y 19** alcanzan la convergencia de la raíz en 4 iteraciones, superando al método de Newton.

Los algoritmos **1, 3, 4, 6, 7, 8, 10, 12 y 15** alcanza la raíz en igual número de iteraciones que el método de Newton, lo cual implica que estos métodos son equivalentes.

El resto de los algoritmos de este capítulo, oscila entre **6 y 8** iteraciones, lo cual significa que no son superiores al método de Newton y que su implementación matemática y computacional no es adecuada y que por tanto, dichos algoritmos no son convenientes para ser utilizado, tanto desde el punto de vista teórico como práctico.

Ejemplo 8.3 *Consideremos la ecuacion no lineal* $e^{-x} + 2\ln x = 0$

Los principales hallazgos respecto a esta ecuación no lineal son :

El método de Newton alcanza la raíz de dicha ecuación en **5** iteraciones, siendo la raíz real con 20 cifras decimales 0,79851808532225987403

Los algoritmos **Ab2, Ab3, 2, 9, 13, 14, 15, 18 y 19** alcanzan la convergencia de la raíz en 4 iteraciones, superando al método de Newton.

Los algoritmos **1, 3, 4, 6, 8, 10, 12 y 16** alcanza la raíz en igual número de iteraciones que el método de Newton, lo cual implica que estos métodos son equivalentes.

El resto de los algoritmos de este capítulo, oscila entre **6 y 7** iteraciones, lo cual significa que no son superiores al método de Newton y que su implementación matemática y computacional no es adecuada y que por tanto, dichos algoritmos no son convenientes para ser utilizado, tanto desde el punto de vista teórico como práctico.

Ejemplo 8.4 *Consideremos la ecuacion no lineal* $x^3 + 4x^2 - 10 = 0$

Los principales hallazgos respecto a esta ecuación no lineal son :

El método de Newton alcanza la raíz de dicha ecuación en **5** iteraciones, siendo la raíz real con 20 cifras decimales 1,36523001341409688791

Los algoritmos **Ab2, Ab3, 2, 9, 13, 14, 18 y 19** alcanzan la convergencia de la raíz en **4** iteraciones, superando al método de Newton.

Los algoritmos **1, 3, 4, 6, 10 y 12** alcanza la raíz en igual número de iteraciones que el método de Newton, lo cual implica que estos métodos son equivalentes.

El resto de los algoritmos de este capítulo, oscila entre **6 y 7** iteraciones, lo cual significa que no son superiores al método de Newton y que su implementación matemática y computacional no es adecuada y que por tanto, dichos algoritmos no son convenientes para ser utilizado, tanto desde el punto de vista teórico como práctico.

Ejemplo 8.5 *Consideremos la ecuacion no lineal* $e^x - x^3 = 0$

Los principales hallazgos respecto a esta ecuación no lineal son :

El método de Newton alcanza la raíz de dicha ecuación en **4** iteraciones, siendo la raíz real con 20 cifras decimales 1,85718386020783521317

Los algoritmos **Ab2, Ab3, 2, 9, 13, 14, 15, 16, 18 y 19** alcanzan la convergencia de la raíz en **3** iteraciones, superando al método de Newton.

Los algoritmos **1, 3, 4, 6, 7, 8, 10, 11 y 12** alcanza la raíz en igual número de iteraciones que el método de Newton, lo cual implica que estos métodos son equivalentes.

El resto de los algoritmos de este capítulo, oscila entre **5 y 9** iteraciones, lo cual significa que no son superiores al método de Newton y que su implementación matemática y computacional no es adecuada y que por tanto, dichos algoritmos no son convenientes para ser utilizado, tanto desde el punto de vista teórico como práctico.

8.6. Conclusiones

Las conclusiones referentes al segundo capítulo se pueden sistematizar en las siguientes aseveraciones :

- Generación de nuevos esquemas iterativos que representan **19** nuevas variantes del método de Newton.
- Existencia de nuevos algoritmos, que constituyen variantes del método de Newton que superan al método clásico en el número de iteraciones necesarias para la obtención de una raíz simple.
- Los nuevos algoritmos obtenidos son aplicaciones directa de haber considerado los métodos de Abbasbandy y Chun, con un desarrollo de la serie de Taylor de orden 2 y 3 respectivamente.
- En el conjunto de funciones base, se muestra la existencia de algoritmos que son equivalente al método de Newton en la cantidad de iteraciones para lograr la raíz real.
- Todos los algoritmos obtenidos por estas variantes del método de Newton son convergente, oscilando todos ellos entre **3 y 9** iteraciones como máximo.
- Extensión de la fórmula de Abbasbandy en los algoritmos Ab2 y Ab3.
- Obtención de diversos polinomios de Adomian para la generación de nuevos métodos iterativos.
- Incorporación de los polinomios de Adomian para la generacion de nuevos métodos iterativos basados en los métodos de Abbasbandy y Chun.

9. CAPÍTULO IV : ITERACIÓN VARIACIONAL

En este capítulo se emplea la técnica iterativa variacional para analizar y construir algunos métodos iterativos que permitan resolver ecuaciones no lineales del tipo $f(x_n) = 0$. Esta técnica permite obtener métodos conocidos del tipo Ostrowsky y muchos casos especiales. También se analizará sus criterios de convergencia y sus principales esquemas iterativos. Al final del mismo, se muestran varios ejemplos donde se analizan la comparación entre dichos métodos.

9.1. Caso I : $H(x) = x + \lambda f(x) g(x)$

Consideremos una función $H(x)$ definida por

$$H(x) = x + \lambda f(x) g(x)$$

donde $g(x)$ es una función arbitraria y λ es el multiplicador de Lagrange.

Derivando respecto a x, obtenemos

$$\begin{aligned} H'(x) &= 1 + \lambda (f(x) g(x))' \\ &= 1 + \lambda (f'(x) g(x) + g'(x) f(x)) \end{aligned}$$

La condición de optimalidad implica que

$$\lambda = -\frac{1}{f'(x) g(x) + g'(x) f(x)}$$

sustituyendo en $H(x) = x + \lambda f(x) g(x)$ y considerando la expresión de punto fijo

$$x = H(x)$$

se tiene el siguiente esquema iterativo principal

$$\begin{cases} x_n = H(x_n) \\ \\ x_{n+1} = x_n - \dfrac{f(x_n) g(x_n)}{f'(x_n) g(x_n) + g'(x_n) f(x_n)} \end{cases}$$

Este algoritmo constituye la principal relación de recurrencia y fue usada por Noor [65] para la obtención de nuevos métodos iterativos para resolver ecuaciones no lineales. También esta fórmula fue obtenida en el capítulo I. A partir de este esquema principal y considerando diversas funciones auxiliares, se encuentran conocidos y nuevos métodos iterativos.

Para precisar dicha idea, consideremos el siguiente algoritmo

Algoritmo 50 *Sea la función auxiliar definida por* $g(x) = x$, *obtenemos el método iterativo*

$$x_{n+1} = x_n - \frac{(x_n) f(x_n)}{(x_n) f'(x_n) + f(x_n)} \tag{79}$$

Algoritmo 51 *Sea la función auxiliar definida por* $g(x) = f'(x)$ *entonces*

$$x_{n+1} = x_n - \frac{f(x_n) f'(x_n)}{[f'(x_n)]^2 + f''(x_n) f(x_n)} \tag{80}$$

Algoritmo 52 *Sea la función auxiliar definida por* $g(x) = f''(x)$ *entonces*

$$x_{n+1} = x_n - \frac{f(x_n) f''(x_n)}{f'(x_n) f''(x_n) + f'''(x_n) f(x_n)} \tag{81}$$

Algoritmo 53 *Sea la función auxiliar definida por* $g(x) = \dfrac{1}{f'(x)}$ *entonces*

$$x_{n+1} = x_n + \frac{f(x_n) f'(x_n)}{f''(x_n) f(x_n) - [f'(x_n)]^2} \tag{82}$$

Algoritmo 54 *Sea la función auxiliar definida por* $g(x) = \dfrac{f(x)}{f'(x)}$ *entonces*

$$x_{n+1} = x_n - \frac{f^2(x_n) f'(x_n)}{2f(x_n)[f'(x_n)]^2 - f^2(x_n) f''(x_n)} \tag{83}$$

Algoritmo 55 *Sea la función auxiliar definida por* $g(x) = e^{-\alpha x}$ *se tiene*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) - \alpha f(x_n)} \tag{84}$$

Notemos que si $\alpha = \dfrac{1}{2}\dfrac{f''(x)}{f'(x)}$ obtenemos otro nuevo método iterativo, definido por

$$x_{n+1} = x_n - \frac{2f(x_n) f'(x_n)}{2[f'(x_n)]^2 - f(x_n) f''(x_n)}$$

el cual constituye el método iterativo de Halley con orden de convergencia cúbica, ver [20, 21, 29].

Algoritmo 56 *Sea la función auxiliar definida por* $g(x) = e^{-\alpha f(x)}$ *entonces*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) - \alpha f'(x_n) f(x_n)} \tag{85}$$

Algoritmo 57 *Sea la función auxiliar definida por* $g(x) = e^{\alpha x}$ *luego*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) + \alpha f(x_n)} \tag{86}$$

Algoritmo 58 *Sea la función auxiliar definida por* $g(x) = e^{\alpha f(x)}$ *, por tanto*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) + \alpha f(x_n) f'(x_n)} \tag{87}$$

Algoritmo 59 *Sea la función auxiliar definida por* $g(x) = e^{-\frac{f(x)}{f'(x)}}$ *en consecuencia*

$$x_{n+1} = x_n - \frac{f(x_n)[f'(x_n)]^2}{[f'(x_n)]^3 - [f'(x_n)]^2 f(x_n) + f''(x_n) f^2(x_n)} \tag{88}$$

Algoritmo 60 *Sea la función auxiliar definida por* $g(x) = e^{-\frac{f''(x)}{f'(x)}}$ *entonces*

$$x_{n+1} = x_n - \frac{f(x_n)[f'(x_n)]^2}{[f'(x_n)]^3 - f'''(x_n) f'(x_n) f(x_n) + [f''(x_n)]^2 f(x_n)} \tag{89}$$

Algoritmo 61 *Sea la función auxiliar definida por* $g(x) = e^{-f'(x)}$ *luego*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n) - f(x_n) f''(x_n)} \tag{90}$$

Algoritmo 62 *Sea la función auxiliar definida por* $g(x) = e^{\alpha/f'(x)}$ *entonces*

$$x_{n+1} = x_n - \frac{f(x_n)[f'(x_n)]^2}{[f'(x_n)]^3 - \alpha f(x_n) f''(x_n)} \tag{91}$$

Notemos que si $\alpha = \frac{1}{2} f'(x)$ obtenemos nuevamente el método iterativo de Halley.

Algoritmo 63 *Sea la función auxiliar definida por* $g(x) = e^{-1/f''(x)}$ *entonces*

$$x_{n+1} = x_n - \frac{f(x_n)[f''(x_n)]^2}{f'(x_n)[f''(x_n)]^2 + f(x_n) f'''(x_n)} \tag{92}$$

Algoritmo 64 *Sea la función auxiliar definida por* $g(x) = e^{-f'(x)/f''(x)}$ *entonces*

$$x_{n+1} = x_n - \frac{f(x_n)[f''(x_n)]^2}{f'(x_n)[f''(x_n)]^2 - [f''(x_n)]^2 f(x_n) + f'''(x_n) f'(x_n) f(x_n)} \tag{93}$$

Ahora vamos a considerar una nueva variante del método de la técnica variacional, constituyendo esto una generalización del método anterior.

9.2. Caso II. $H_1(x) = \phi(x) + \lambda (f(x) g(x))^p$, $p = 1$

Sea

$$H_1(x) = \phi(x) + \lambda (f(x) g(x))^p$$

donde $\phi(x)$ es una función iterativa de orden $p \geq 1$, $g(x)$ una función arbitraria y λ el Multiplicador de Lagrange.

La condición de optimalidad es

$$\lambda = -\frac{\phi'(x)}{p (f(x) g(x))^{p-1} (f(x) g(x))'}$$

sustituyendo en $H(x) = \phi(x) + \lambda (f(x) g(x))^p$, el esquema iterativo principal se convierte en

$$H_1(x) = \phi(x) - \frac{\phi'(x) (f(x) g(x))}{p (f'(x) g(x) + g'(x) f(x))}$$

ahora, para $p = 1$, obtenemos

$$H_1(x_n) = \phi(x_n) - \frac{\phi'(x_n) f(x_n) g(x_n)}{f'(x_n) g(x_n) + g'(x_n) f(x_n)}$$

Consideremos la función iterativa $\phi(x) = x - \dfrac{f(x)}{f'(x)}$ (Método de Newton), derivando esta función

$$\phi'(x) = \frac{f(x) f''(x)}{[f'(x)]^2}$$

entonces, el principal esquema iterativo se convierte en

$$H_1(x_n) = \phi(x_n) - \frac{\phi'(x_n) (f(x_n) g(x_n))}{(f(x_n) g(x_n))'}$$

Notemos que si $\phi(x) = x$ se obtiene el método iterativo implementado en el capítulo I, es decir

$$x_{n+1} = x_n - \frac{f(x_n) g(x_n)}{(f(x_n) g(x_n))'}$$

Ahora considerando la función $\phi(x_n)$ junto con su derivada $\phi'(x_n)$ y sustituyendo en la forma funcional $H_1(x_n)$, obtenemos un nuevo método iterativo, definido por

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{[f(x_n)]^2 f''(x_n) g(x_n)}{[f'(x_n)]^2 (f'(x_n) g(x_n) + g'(x_n) f(x_n))} \tag{94}$$

Ahora vamos a obtener nuevos métodos iterativos, seleccionando adecuadamente funciones $g(x_n)$ que permitan simplificar dicha expresión iterativa. Consideremos diversas funciones arbitrarias.

Algoritmo 65 *Sea la función auxiliar $g(x) = 1$ entonces el esquema iterativo anterior , toma la forma*

$$\left\{ \begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^2(x_n) f''(x_n)}{[f'(x_n)]^3} \end{array} \right. \tag{95}$$

Algoritmo 66 *Sea la función auxiliar $g(x) = x$ entonces el método iterativo toma la forma*

$$\left\{ \begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f(x_n)]^2 f''(x_n)(x_n)}{[f'(x_n)]^2 ((x_n) f'(x_n) + f(x_n))} \end{array} \right. \tag{96}$$

Algoritmo 67 *Sea la función auxiliar $g(x) = f(x)$ entonces el método iterativo anterior es*

$$\left\{ \begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f(x_n)]^3 f''(x_n)}{2[f'(x_n)]^2 f'(x_n) f(x_n)} \end{array} \right. \tag{97}$$

Algoritmo 68 *Sea $g(x) = f'(x)$ entonces el esquema iterativo anterior, toma la forma*

$$\left\{ \begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f(x_n)]^2 f''(x_n) f'(x_n)}{[f'(x_n)]^4 + f''(x_n)[f'(x_n)]^2 f(x_n)} \end{array} \right. \tag{98}$$

Consideremos ahora la familia de funciones exponenciales y algunas de sus principales variantes:

Algoritmo 69 *Sea $g(x) = e^{-\alpha x}$ entonces el nuevo método iterativo es*

$$\left\{ \begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f(x_n)]^2 f''(x_n)}{[f'(x_n)]^2 (f'(x_n) - \alpha f(x_n))} \end{array} \right. \tag{99}$$

Algoritmo 70 *Si $\alpha = 0$, el esquema iterativo toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f(x_n)]^2 f''(x_n)}{[f'(x_n)]^3} \end{cases} \tag{100}$$

Algoritmo 71 *Sea $g(x) = f(x)e^{-\alpha x}$ entonces el nuevo método iterativo es*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f(x_n)]^2 f''(x_n)}{[f'(x_n)]^2 [2f'(x_n) - \alpha f(x_n)]} \end{cases} \tag{101}$$

Algoritmo 72 *Si $\alpha = 0$, el esquema iterativo anterior toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f(x_n)]^2 f''(x_n)}{2[f'(x_n)]^3} \end{cases} \tag{102}$$

Algoritmo 73 *Sea $g(x) = e^{-\alpha f(x)}$ entonces $g'(x) = -\alpha e^{-\alpha f(x)}$ y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f(x_n)]^2 f''(x_n)}{[f'(x_n)]^2 (f'(x_n) - \alpha f'(x_n) f(x_n))} \end{cases} \tag{103}$$

En la fórmula anterior, notemos que si $\alpha = 0$, el esquema iterativo toma la forma

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f(x_n)]^2 f''(x_n)}{[f'(x_n)]^3} \end{cases} \tag{104}$$

Algoritmo 74 *Sea $g(x) = f(x)e^{-\alpha f(x)}$ entonces $g'(x) = f'(x)e^{-\alpha f(x)} - \alpha e^{-\alpha f(x)} f'(x) f(x)$, luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f(x_n)]^2 f''(x_n)}{[f'(x_n)]^2 (f'(x_n) f(x_n) + f'(x_n) - \alpha f'(x_n) f(x_n))} \end{cases} \tag{105}$$

Notemos que si $\alpha = 1$, el esquema iterativo toma la forma

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f(x_n)]^2 f''(x_n)}{[f'(x_n)]^3} \end{cases}$$

Consideremos ahora algunos cocientes funcionales,

Algoritmo 75 *Sea* $g(x) = \dfrac{f(x)}{f'(x)}$ *entonces* $g'(x) = -\dfrac{[f'(x)]^2 - f''(x) f(x)}{[f'(x)]^2}$ *luego el algoritmo es*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f(x_n)]^3 f''(x_n) f'(x_n)}{[f'(x_n)]^4 f(x_n) - [f'(x_n)]^2 f(x_n) - f''(x_n) f^2(x_n)} \end{cases} \tag{106}$$

Algoritmo 76 *Sea* $g(x) = \dfrac{1}{2}\dfrac{f''(x)}{f'(x)}$ *entonces* $g'(x) = \dfrac{1}{2}\dfrac{f'''(x) f'(x) - [f''(x)]^2}{[f'(x)]^2}$, *de donde*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f(x_n)]^2 f''(x_n)}{[f'(x_n)]^3 \left(2f''(x_n) + f'''(x_n) f'(x_n) - [f''(x_n)]^2\right)} \end{cases} \tag{107}$$

Algoritmo 77 *Sea* $g(x) = \dfrac{1}{f(x)}$ *entonces* $g'(x) = -\dfrac{f'(x)}{[f(x)]^2}$, *luego el algoritmo es*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f(x_n)]^2 f''(x_n)}{[f'(x_n)]^3 - f(x_n) f'(x_n) f''(x_n)} \end{cases} \tag{108}$$

Algoritmo 78 *Sea* $g(x) = \dfrac{1}{f^2(x)}$ *entonces* $g'(x) = -\dfrac{2f'(x)}{[f(x)]^3}$, *y el esquema iterativo principal toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n + \dfrac{[f(x_n)]^2 f''(x_n)}{[f'(x_n)]^3 - f(x_n) f'(x_n) f''(x_n)} \end{cases} \tag{109}$$

Algoritmo 79 *Sea* $g(x) = \dfrac{f''(x)}{f'(x)}$ *entonces* $g'(x) = \dfrac{f'''(x) f'(x) - [f''(x)]^2}{[f'(x)]^2}$ *y el esquema iterativo anterior, toma la forma*

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f(x_n)]^2 [f''(x_n)]^2}{[f'(x_n)]^4 f''(x_n) - f'''(x_n)[f'(x_n)]^3 f(x_n) - f(x_n)[f'(x_n) f''(x_n)]^2} \end{array}\right. \tag{110}$$

Algoritmo 80 *Sea* $g(x) = \dfrac{1}{\phi(x)}$ *entonces* $g'(x) = -\dfrac{\phi'(x)}{[\phi(x)]^2}$, *de donde*

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n + \dfrac{\left[[f(x_n)]^2 f''(x_n)\right][x f'(x_n) - f(x_n)]}{f^2(x_n) f''(x_n) f'(x_n) + f(x_n)[f'(x_n)]^3 - x_n [f'(x_n)]^4} \end{array}\right. \tag{111}$$

Algoritmo 81 *Sea* $g(x) = \dfrac{1}{\phi'(x)}$ *entonces* $g'(x) = -\dfrac{\phi''(x)}{[\phi'(x)]^2}$, *luego*

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^4(x_n)(f''(x_n))^4}{(f'(x_n))^7 f''(x_n) + 2f(x_n) f'(x_n)(f''(x_n))^2 - f'(x_n) f'(x_n) f''(x_n) - f(x_n) f'''(x_n) f'(x_n)} \end{array}\right. \tag{112}$$

9.3. Caso III : $H(x) = \phi(x) - \dfrac{\phi'(x)(f(x)g(x))}{(f(x)g(x))'}$

Ahora vamos a considerar otra variante del esquema iterativo

$$H(x) = \phi(x) - \frac{\phi'(x)(f(x)g(x))}{(f(x)g(x))'}$$

donde su método iterativo está expresado por

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{[f(x_n)]^2 f''(x_n) g(x_n)}{[f'(x_n)]^2 (f'(x_n) g(x_n) + g'(x_n) f(x_n))}$$

Hagamos el desarrollo de la serie de Taylor de orden 2 de la función $f(y)$ alrededor del punto x, es decir

$$f(y) \approx f(x) + (y-x) f'(x) + \frac{(y-x)^2}{2} f''(x)$$

donde

$$y = x - \frac{f(x)}{f'(x)}$$

entonces de las dos expresiones anteriores, se tiene que

$$f(y) = \frac{f^2(x) f''(x)}{2[f'(x)]^2}$$

de donde

$$f''(x) = \frac{2f(y)[f'(x)]^2}{f^2(x)}$$

y el esquema iterativo, toma la forma

$$x_n = y_n - \frac{f(y_n) g(x_n)}{(f'(x_n) g(x_n) + g'(x_n) f(x_n))} \tag{113}$$

Otra nueva variante, surge cuando se considera a la función iterativa

$$\phi(x) = x - \frac{f(x)}{f'(x) - \alpha f(x)}$$

entonces

$$\phi'(x) = 1 - \frac{[f'(x)]^2 - f(x) f''(x)}{[f'(x) - \alpha f(x)]^2}$$

ahora consideremos la función auxiliar $g(x) = e^{-\alpha x}$ entonces obtenemos un nuevo método iterativo definido por

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n) - \alpha f(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n)\left[1 - \dfrac{[f'(x_n)]^2 - f(x_n) f''(x_n)}{[f'(x_n) - \alpha f(x_n)]^2}\right]}{2[f'(x_n) - \alpha f(x_n)]} \end{cases} \tag{114}$$

Ahora vamos a considerar el método iterativo presentado en [56], donde la función iterativa es

$$\phi(x) = x - \frac{f(x) f'(x)}{[f(x)]^2 + [f'(x)]^2}$$

derivando esta función, se tiene

$$\phi'(x) = 1 - \frac{\left[f(x) f''(x) - [f'(x)]^2\right]\left[[f^2(x)] - [f'(x)]^2\right]}{\left[[f'(x)]^2 + f^2(x)\right]^2}$$

y de esta manera, obtenemos el nuevo esquema iterativo

$$x_{n+1} = x_n - \frac{f(x_n) f'(x_n)}{[f(x_n)]^2 + [f'(x_n)]^2} - \frac{f(x_n)}{2[f'(x_n) - f(x_n)]}\left[1 - \frac{\left[f(x_n) f''(x_n) - [f'(x_n)]^2\right]\left[[f^2(x_n)] - [f'(x_n)]^2\right]}{\left[[f'(x_n)]^2 + f^2(x_n)\right]^2}\right] \tag{115}$$

ahora sustituyendo la segunda derivada por una diferencia finita del tipo

$$f''(x) \approx \frac{f'(y) - f'(x)}{y - x}$$

y sustituyendo en el esquema iterativo la expresión anterior

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f''(x_n) f^2(x_n)}{2[f'(x_n)]^2 [f'(x_n) - \alpha f(x_n)]}$$

obtenemos el nuevo método iterativo, dado por

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f(x_n)}{2f'(x_n)} \frac{f'(x_n) - f'(y_n)}{f'(x_n) - \alpha f(x_n)} \tag{116}$$

Ahora vamos a considerar otra nueva variante del esquema iterativo

$$H(x) = \phi(x) - \frac{\phi'(x)(f(x) g(x))}{(f(x) g(x))'}$$

Tomemos a la función iterativa $\phi(x)$ como la función de Traub [94], es decir

$$\phi(x) = y - \frac{f(y)}{f'(y)}$$

donde

$$y = x - \frac{f(x)}{f'(x)}$$

derivando, se tiene que

$$\phi'(x) = \frac{f(y) f''(y)}{[f'(y)]^2} y'$$

además

$$y' = \frac{f(x) f''(x)}{[f'(x)]^2}$$

obtenemos el nuevo método iterativo

$$x_{n+1} = y_n - \frac{f(y_n)}{f'(y_n)} - \frac{f(y_n) f''(y_n)}{[f'(x_n)]^2} \frac{f(x_n) f''(x_n)}{[f'(x_n)]^2}\left[\frac{f(x_n) g(x_n)}{p[f'(x_n) g(x_n) + g'(x_n) f(x_n)]}\right] \tag{117}$$

Finalmente vamos a considerar otra nueva variante del esquema iterativo

$$H(x_n) = \phi(x_n) - \frac{\phi'(x_n)(f(x_n) g(x_n))}{(f(x_n) g(x_n))'}$$

Hagamos el desarrollo de la serie de Taylor de orden 2 de la función $f(z)$ alrededor del punto y, es decir

$$f(z) \approx f(y) + (z-y) f'(y) + \frac{(z-y)^2}{2} f''(y)$$

donde

$$z = y - \frac{f(y)}{f'(y)}$$

entonces de las dos expresiones anteriores, se tiene que

$$f(z) = \frac{f^2(y) f''(y)}{2[f'(y)]^2}$$

de donde

$$f''(y) = \frac{2f(z)[f'(y)]^2}{f^2(y)}$$

y el esquema iterativo, toma la forma

$$x_{n+1} = z_n - \frac{f(z_n) g(x_n)}{(f'(x_n) g(x_n) + g'(x_n) f(x_n))}$$

y entonces el nuevo método iterativo obtenido es de la forma

$$\left\{ \begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ z_n = y_n - \dfrac{f(y_n)}{f'(y_n)} \\ \\ x_{n+1} = z_n - \dfrac{f(z_n) g(x_n)}{(f'(x_n) g(x_n) + g'(x_n) f(x_n))} \end{array} \right. \tag{118}$$

Vamos a considerar a la función iterativa $\phi(x)$ como la función de Traub [94], es decir

$$\phi(x) = y - \frac{f(y)}{f'(y)}$$

donde

$$y = x - \frac{f(x)}{f'(x)}$$

derivando, se tiene que

$$\phi'(x) = \frac{f(y) f''(y)}{[f'(y)]^2} y'$$

además

$$y' = \frac{f(x) f''(x)}{[f'(x)]^2}$$

obtenemos el nuevo método iterativo

$$x_{n+1} = y_n - \frac{f(y_n)}{f'(y_n)} - \frac{f(y_n) f''(y_n)}{[f'(x_n)]^2} \frac{f(x_n) f''(x_n)}{[f'(x_n)]^2} \left[\frac{f(x_n) g(x_n)}{p [f'(x_n) g(x_n) + g'(x_n) f(x_n)]} \right] \tag{119}$$

Ahora vamos a considerar otra variante del método de la técnica iterativa variacional y se obtiene cuando se sustituye la segunda derivada en función de la primer derivada y el valor de p se considera igual a 2. Para precisar, consideremos el siguiente caso.

9.4. Caso IV : $H(x) = \phi(x) - \dfrac{\phi'(x)(f(x) g(x))}{p(f'(x) g(x) + g'(x) f(x))}$

Hagamos el desarrollo de la función $f(y)$ en serie de Taylor de orden 2, alrededor de x, es decir

$$f(y) = f(x) + f'(x)(y - x) + \frac{f''(x)}{2}(y - x)^2$$

haciendo

$$y = x - \frac{f(x)}{f'(x)}$$

tenemos que

$$f(y) = \frac{f^2(x) f''(x)}{2(f'(x))^2}$$

de donde

$$f''(x) = \frac{2f(y)(f'(x))^2}{f^2(x)}$$

y sustituyendo en el esquema iterativo principal anterior (Caso II), llegamos a un nuevo esquema iterativo dado por

$$H(x_n) = \phi(x_n) - \frac{\phi'(x_n)(f(x_n) g(x_n))}{p(f'(x_n) g(x_n) + g'(x_n) f(x_n))}$$

con un valor de $p = 2$, obtenemos

$$x_n = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f(x_n) g(x_n) f''(x_n)}{[f'(x_n)]^2 [f'(x_n) g(x_n) + g'(x_n) f(x_n)]}$$

y de esta manera, hemos obtenidos un nuevo esquema iterativo, dado por

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n) g(x_n)}{[f'(x_n) g(x_n) + g'(x_n) f(x_n)]} \end{cases} \tag{120}$$

Podemos notar la analogía entre los esquemas iterativos de los casos II y del presente caso, donde la función $f(x)$ del numerador ha sido sustituida por la función $f(y)$, donde $y = x - \dfrac{f(x)}{f'(x)}$, es decir y constituye una función predictora que en este caso es el método clásico de Newton. Por otro lado, la estructura del esquema iterativo es invariante y la característica más notable de este nuevo esquema iterativo es que se hace uso de una composición de función, es decir

$$f(y) = f\left(x - \frac{f(x)}{f'(x)}\right)$$

Para generalizar el caso II, nuevamente vamos a considerar algunas variantes de la función auxiliar $g(x)$ con el objetivo de obtener nuevos métodos iterativos.

Algoritmo 82 *Sea $g(x) = 1$ entonces $g'(x) = 0$, luego*

$$\left\{ \begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n)} \end{array} \right. \tag{121}$$

Algoritmo 83 *Sea $g(x) = \alpha x$ entonces $g'(x) = \alpha$ y el método iterativo es*

$$\left\{ \begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{(x_n)\, f(y_n)}{[(x_n)\, f'(x_n) + f(x_n)]} \end{array} \right. \tag{122}$$

Algoritmo 84 *Sea $g(x) = f(x)$ entonces $g'(x) = f'(x)$, y el algoritmo es*

$$\left\{ \begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{2f'(x_n)} \end{array} \right. \tag{123}$$

Algoritmo 85 *Sea $g(x) = f'(x)$ entonces el esquema iterativo anterior, toma la forma*

$$\left\{ \begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)\, f'(x_n)}{[f'(x_n)]^2 + f''(x_n)\, f(x_n)} \end{array} \right. \tag{124}$$

Algoritmo 86 *Sea* $g(x) = f''(x)$ *entonces el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ x_{n+1} = y_n - \dfrac{f(y_n) f''(x_n)}{f'(x_n) f''(x_n) + f'''(x_n) f(x_n)} \end{cases} \tag{125}$$

Algoritmo 87 *Sea* $g(x) = f(y)$ *entonces* $g'(x) = f'(y) f(x) f''(x)$ *, y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^2(y_n)}{[f'(x_n) f(y_n) + f'(y_n) f^2(x_n) f''(x_n)]} \end{cases} \tag{126}$$

Algoritmo 88 *Sea* $g(x) = f'(y)$ *entonces* $g'(x) = f''(y) f(x) f''(x)$ *, y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n) f'(y_n)}{[f'(x_n) f'(y_n) + f''(y_n) f^2(x_n) f''(x_n)]} \end{cases} \tag{127}$$

Algoritmo 89 *Sea* $g(x) = \dfrac{1}{f(y)}$ *entonces* $g'(x) = -\dfrac{f''(y) f^2(x) f''(x)}{f^2(y)}$ *, y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ x_{n+1} = y_n - \dfrac{f^2(y)}{f'(x_n) f(y_n) - f''(y_n) f^2(x_n) f''(x_n)} \end{cases} \tag{128}$$

Algoritmo 90 *Sea* $g(x) = \dfrac{1}{f^2(y)}$ *entonces* $g'(x) = -\dfrac{2f''(y) f^2(x) f''(x)}{f^3(y)}$ *, y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^2(y_n)}{f'(x_n) f(y_n) - 2f''(y_n) f^3(x_n) f''(x_n)} \end{cases} \tag{129}$$

Algoritmo 91 *Sea* $g(x) = e^{-\alpha x}$ *entonces* $g'(x) = -\alpha e^{-ax}$ *, y el esquema iterativo anterior, toma la forma*

$$\left\{ \begin{array}{r} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n) - \alpha f(x_n)} \end{array} \right. \tag{130}$$

Algoritmo 92 *Sea* $g(x) = e^{-\alpha f(x)}$ *entonces* $g'(x) = -\alpha f'(x) e^{-af(x)}$ *, y el esquema iterativo anterior, toma la forma*

$$\left\{ \begin{array}{r} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n) - \alpha f(x_n) f'(x_n)} \end{array} \right. \tag{131}$$

Algoritmo 93 *Sea* $g(x) = f(x) e^{-\alpha x}$ *entonces* $g'(x) = f'(x) e^{-\alpha x} - \alpha e^{-ax} f(x)$ *, y el esquema iterativo anterior, toma la forma*

$$\left\{ \begin{array}{r} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{2f'(x_n) - \alpha f(x_n)} \end{array} \right. \tag{132}$$

Algoritmo 94 *Sea* $g(x) = f(x) e^{-\alpha f(x)}$ *entonces* $g'(x) = f'(x) e^{-\alpha f(x)} - \alpha f'(x) e^{-af(x)} f(x)$ *, y el esquema iterativo anterior, toma la forma*

$$\left\{ \begin{array}{r} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{2f'(x_n) - \alpha f(x_n) f'(x_n)} \end{array} \right. \tag{133}$$

Algoritmo 95 *Sea* $g(x) = e^{\alpha f(x)}$ *entonces* $g'(x) = \alpha f'(x) e^{af(x)}$ *, y el esquema iterativo anterior, toma la forma*

$$\left\{ \begin{array}{r} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n) + \alpha f(x_n)} \end{array} \right. \tag{134}$$

Algoritmo 96 *Sea* $g(x) = f(x)e^{\alpha x}$ *entonces* $g'(x) = f'(x)e^{\alpha x} + \alpha e^{ax}f(x)$ *, y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{2f'(x_n) + \alpha f(x_n)} \end{cases} \tag{135}$$

Algoritmo 97 *Sea* $g(x) = e^{\frac{\alpha}{f(x)}}$ *entonces* $g'(x) = -\alpha \dfrac{f'(x)}{f^2(x)} e^{\frac{\alpha}{f(x)}}$ *, y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)f^2(x_n)}{f'(x_n)f^2(x_n) - \alpha f'(x_n)f(x_n)} \end{cases} \tag{136}$$

Algoritmo 98 *Sea* $g(x) = \dfrac{1}{f'(y)}$ *entonces* $g'(x) = -\dfrac{f''(y)f^2(x)f''(x)}{[f'(y)]^2}$ *, y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)f'(y_n)}{f'(x_n)f'(y_n) - f''(y)f^3(x_n)f''(x_n)} \end{cases} \tag{137}$$

Algoritmo 99 *Sea* $g(x) = \dfrac{1}{f'(x)}$ *entonces* $g'(x) = -\dfrac{f''(x)}{[f'(x)]^2}$ *, y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)f'(x_n)}{[f'(x_n)]^2 - f(x_n)f''(x_n)} \end{cases} \tag{138}$$

Algoritmo 100 *Sea* $g(x) = \dfrac{f''(x)}{f'(x)}$ *entonces* $g'(x) = \dfrac{f'''(x)f'(x) - [f''(x)]^2}{[f'(x)]^2}$ *, y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)f(x_n)f'(x_n)}{f(x_n)[f'(x_n)]^2 - 2f(y)(f'(x_n))^2} \end{cases} \tag{139}$$

Algoritmo 101 *Sea* $g(x) = e^{\frac{f(x)}{f'(x)}}$ *entonces* $g'(x) = \left(1 - \frac{[f''(x)]^2}{[f'(x)]^2}\right) e^{\frac{f(x)}{f'(x)}}$ *, y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)[f'(x_n)]^2}{[f'(x_n)]^3 + f(x_n)[f'(x_n)]^2 - f(x_n)[f''(x_n)]^2} \end{cases} \tag{140}$$

Algoritmo 102 *Sea* $g(x) = e^{\frac{1}{f'(x)}}$ *entonces* $g'(x) = -\frac{f''(x)}{[f'(x)]^2} e^{\frac{1}{f'(x)}}$ *, y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)[f'(x_n)]^2}{[f'(x_n)]^3 - f(x_n) f''(x_n)} \end{cases} \tag{141}$$

Algoritmo 103 *Sea* $g(x) = e^{\frac{1}{f''(x)}}$ *entonces* $g'(x) = -\frac{f'''(x)}{[f''(x)]^2} e^{\frac{1}{f''(x)}}$ *, y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)[f''(x_n)]^2}{f'(x_n)[f''(x_n)]^2 - f'''(x_n) f(x_n)} \end{cases} \tag{142}$$

Algoritmo 104 *Sea* $g(x) = e^{\alpha f'(x)}$ *entonces* $g'(x) = \alpha f''(x) e^{\alpha f'(x)}$ *, y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n) + \alpha f''(x_n) f(x_n)} \end{cases} \tag{143}$$

Ahora vamos a seguir estudiando otras variantes del caso III con el objeto de poder generalizarlo y obtener nuevos métodos iterativos.

Consideremos la función iterativa $\phi(x) = x - \frac{f(x)}{f'(x)}$ derivando esta función, se obtiene

$$\phi'(x) = 1 - \frac{[f'(x)]^2 - f''(x) f(x)}{[f'(x)]^2}$$

y sustituyendo en el esquema iterativo principal anterior (Caso II)

$$H(x) = \phi(x) - \frac{\phi'(x) f(x) g(x)}{p(f'(x) g(x) + g'(x) f(x))}$$

con un valor de $p = 1$, obtenemos un nuevo esquema iterativo, definido por

$$\begin{cases} y_n = x_n - \dfrac{f(x)}{f'(x)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x) f^2(x) g(x)}{[f'(x)]^2 [f'(x) g(x) + g'(x) f(x)]} \end{cases} \tag{144}$$

Nuevamente vamos a considerar algunas variantes de la función auxiliar $g(x)$ con el objetivo de obtener nuevos métodos iterativos.

Algoritmo 105 *Sea $g(x) = 1$, derivando $g'(x) = 0$ y de esta manera, sustituyendo en la expresión anterior, obtenemos un nuevo método iterativo, dado por*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{[f'(x_n)]^3} \end{cases} \tag{145}$$

Algoritmo 106 *Sea $g(x) = f(x)$, derivando $g'(x) = f'(x)$ y de esta manera, sustituyendo en la expresión anterior, obtenemos un nuevo método iterativo, dado por*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{2[f'(x_n)]^3} \end{cases} \tag{146}$$

Algoritmo 107 *Sea $g(x) = f'(x)$, derivando $g'(x) = f''(x)$ y de esta manera, sustituyendo en la expresión anterior, obtenemos un nuevo método iterativo, dado por*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{f'(x_n) \left([f'(x)]^2 + f(x_n) f''(x_n)\right)} \end{cases} \tag{147}$$

Algoritmo 108 *Sea* $g(x) = \dfrac{1}{f''(x)}$, *derivando* $g'(x) = -\dfrac{f'''(x)}{[f''(x)]^2}$ *y de esta manera, sustituyendo en la expresión anterior, obtenemos un nuevo método iterativo, dado por*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f''(x_n)]^2 f^2(x_n)}{[f'(x_n)]^2 (f'(x) f''(x_n) - f(x_n) f'''(x_n))} \end{cases} \tag{148}$$

Algoritmo 109 *Sea* $g(x) = \dfrac{1}{f^2(x)}$, *derivando* $g'(x) = -\dfrac{2f'(x)}{f^3(x)}$ *y de esta manera, sustituyendo en la expresión anterior, obtenemos un nuevo método iterativo, dado por*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x)}{[f'(x_n)]^2 (f(x) - 2f'(x_n))} \end{cases} \tag{149}$$

Algoritmo 110 *Sea* $g(x) = \dfrac{1}{f''(x) f^2(x)}$, *derivando* $g'(x) = -\dfrac{f'''(x) f^2(x) - 2f(x) f'(x) f''(x)}{[f''(x)]^2 f^4(x)}$ *y de esta manera, sustituyendo en la expresión anterior, obtenemos un nuevo método iterativo, dado por*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f''(x_n)]^2 f^3(x_n)}{[f'(x_n)]^2 (f''(x_n) f'(x_n) f(x_n) - f'''(x_n) f'(x_n) + 2f''(x_n) f'(x_n))} \end{cases} \tag{150}$$

Algoritmo 111 *Sea* $g(x) = e^{-\alpha x}$, *derivando* $g'(x) = -\alpha e^{-\alpha x}$ *y de esta manera, sustituyendo en la expresión anterior, obtenemos un nuevo método iterativo, dado por*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{[f'(x_n)]^2 (f'(x_n) - \alpha f(x_n))} \end{cases} \tag{151}$$

Notemos que si $\alpha = 0$, el método iterativo se reduce a

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{[f'(x_n)]^3} \end{cases} \tag{152}$$

Algoritmo 112 *Sea $g(x) = e^{-\alpha f(x)}$, derivando $g'(x) = -\alpha f'(x) e^{-\alpha f(x)}$ y de esta manera, sustituyendo en la expresión anterior, obtenemos un nuevo método iterativo, dado por*

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{[f'(x_n)]^2 [f'(x_n) - \alpha f(x_n) f'(x_n)]} \end{array}\right. \tag{153}$$

Notemos que si $\alpha = 0$, el método iterativo se reduce a

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x) f^2(x_n)}{[f'(x_n)]^3} \end{array}\right.$$

Algoritmo 113 *Sea $g(x) = f(x) e^{-\alpha x}$, derivando $g'(x) = f'(x) e^{-\alpha x} - \alpha e^{-\alpha x} f(x)$ y de esta manera, sustituyendo en la expresión anterior, obtenemos un nuevo método iterativo, dado por*

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{[f'(x_n)]^2 [2f'(x_n) - \alpha f(x_n)]} \end{array}\right.$$

Notemos que si $\alpha = 0$, el método iterativo se reduce a

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x) f^2(x_n)}{2[f'(x_n)]^3} \end{array}\right.$$

Algoritmo 114 *Sea $g(x) = f(x) e^{-\alpha f(x)}$, derivando $g'(x) = f'(x) e^{-\alpha f(x)} - \alpha f'(x) e^{-\alpha f(x)} f(x)$ y de esta manera, sustituyendo en la expresión anterior, obtenemos un nuevo método iterativo, dado por*

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{[f'(x_n)]^2 [2f'(x_n) - \alpha f'(x_n) f(x_n)]} \end{array}\right.$$

Notemos que si $\alpha = 0$, el método iterativo se reduce a

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{2\left[f'(x_n)\right]^3} \end{cases} \tag{154}$$

Algoritmo 115 *Sea $g(x) = e^{-\frac{\alpha}{f(x)}}$, derivando $g'(x) = -\dfrac{\alpha f'(x)}{f^2(x)} e^{-\frac{\alpha}{f(x)}}$ y de esta manera, sustituyendo en la expresión anterior, obtenemos un nuevo método iterativo, dado por*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^3(x_n)}{\left[f'(x_n)\right]^2 \left[f'(x_n) f(x_n) - \alpha f'(x_n)\right]} \end{cases}$$

Algoritmo 116 *Notemos que si $\alpha = 0$, el método iterativo se reduce a*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{\left[f'(x_n)\right]^3} \end{cases} \tag{155}$$

Algoritmo 117 *Sea $g(x) = e^{-\alpha \frac{f(x)}{f'(x)}}$, derivando $g'(x) = -\alpha \dfrac{\left[f'(x)\right]^2 - f''(x) f(x)}{\left[f'(x)\right]^2} e^{-\alpha \frac{f(x)}{f'(x)}}$ y de esta manera, sustituyendo en la expresión anterior, obtenemos un nuevo método iterativo, dado por*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{\left[f'(x_n)\right]^3 - \alpha f(x_n) \left[f'(x_n)\right]^2 + \alpha f^2(x_n) f''(x_n)} \end{cases}$$

Notemos que si $\alpha = 0$, el método iterativo se reduce a

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{\left[f'(x_n)\right]^3} \end{cases} \tag{156}$$

Algoritmo 118 *Sea* $g(x) = e^{-\alpha \frac{f'(x)}{f''(x)}}$, *derivando* $g'(x) = -\alpha \dfrac{[f''(x)]^2 - f'''(x) f'(x)}{[f''(x)]^2} e^{-\alpha \frac{f'(x)}{f''(x)}}$ *y de esta manera, sustituyendo en la expresión anterior, obtenemos un nuevo método iterativo, dado por*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f''(x_n)]^3 f^2(x_n)}{[f'(x_n)]^2 \left[f'(x_n)[f''(x_n)]^2 - \alpha f(x_n)[f''(x_n)]^2 + \alpha f(x_n) f'(x_n) f'''(x_n)\right]} \end{cases}$$

Algoritmo 119 *Si* $\alpha = 0$, *el método iterativo se reduce a*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f''(x_n)]^3 f^2(x_n)}{[f'(x_n)]^3 [f''(x_n)]^2} \end{cases}$$

9.5. Caso V : $H(x) = \phi(x) + \lambda (f(x) g(x))^p$, $p = 2$

Sea la forma funcional

$$H(x) = \phi(x) + \lambda (f(x) g(x))^p$$

donde

$\phi(x)$: Función iterativa de orden $p \geq 1$

$g(x)$: Función arbitraria

λ : Multiplicador de Lagrange

Por otro lado, la condición de optimalidad implica que

$$\lambda = -\frac{\phi'(x)}{p(fg)^{p-1}(fg)'}$$

entonces el esquema iterativo principal se convierte en

$$H(x_n) = \phi(x_n) - \frac{\phi'(x_n) f(x_n) g(x_n)}{p(f'(x_n) g(x_n) + g'(x_n) f(x_n))}$$

Notemos que para $\phi(x_n) = x_n$ y $p = 1$, obtenemos

$$H(x_n) = x_n - \frac{f(x_n) g(x_n)}{f'(x_n) g(x_n) + g'(x_n) f(x_n)}$$

fórmula que fue obtenida en el algoritmo dedicado al método de Newton.

Igual que en el caso anterior, consideremos la función iterativa definida por el método de Newton

$$\phi(x) = x - \frac{f(x)}{f'(x)}$$

derivando esta función

$$\phi'(x) = \frac{f(x) f''(x)}{[f'(x)]^2}$$

entonces, para $p = 2$, se tiene

$$\begin{cases} H(x_n) = \phi(x_n) - \dfrac{\phi'(x_n) f(x_n) g(x_n)}{p(f'(x_n) g(x_n) + g'(x_n) f(x_n))} \\ \\ H(x_n) = y_n - \dfrac{f''(x_n) f^2(x_n) g(x_n)}{2[f'(x_n)]^2 (f'(x_n) g(x_n) + g'(x_n) f(x_n))} \end{cases} \tag{157}$$

Ahora vamos a obtener nuevos métodos iterativos, seleccionando adecuadamente funciones $g(x)$ que permitan simplificar dicha expresión iterativa. Consideremos diversas funciones arbitrarias.

Algoritmo 120 *Sea la función $g(x) = 1$ entonces el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{2[f'(x_n)]^3} \end{cases} \tag{158}$$

Algoritmo 121 *Sea la función auxiliar $g(x) = x$ entonces el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{(x_n) f''(x_n) f^2(x_n)}{2[f'(x_n)]^2 [(x_n) f'(x_n) + f(x_n)]} \end{cases} \tag{159}$$

Algoritmo 122 *Sea $g(x) = \alpha f(x)$ entonces $g'(x) = \alpha f'(x)$ y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{4[f'(x_n)]^3} \end{cases} \tag{160}$$

Algoritmo 123 *Sea la función auxiliar $g(x) = f^2(x)$ entonces $g'(x) = 2f(x) f'(x)$ y el esquema iterativo anterior, toma la forma*

$$\left\{ \begin{array}{r} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{6 [f'(x_n)]^3} \end{array} \right. \tag{161}$$

Notemos que si $g(x) = [f(x)]^p$, $p \in \mathbb{Z}^+$ entonces $g'(x) = p [f(x)]^{p-1} f'(x)$ y el esquema iterativo anterior, toma la forma

$$\left\{ \begin{array}{r} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ H(x_n) = x_n - \dfrac{f(x_n)}{f'(x_n)} - \dfrac{[f(x_n)]^2 f''(x_n)}{(2+2p) [f'(x_n)]^{p+1}} \end{array} \right. \tag{162}$$

Consideremos algunas variantes de las derivadas,

Algoritmo 124 *Sea $g(x) = f'(x)$ entonces $g'(x) = f''(x)$ y el esquema iterativo anterior, toma la forma*

$$\left\{ \begin{array}{r} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n) f'(x_n)}{2 [f'(x_n)]^2 \left([f'(x_n)]^2 + f''(x_n) f(x_n) \right)} \end{array} \right. \tag{163}$$

Algoritmo 125 *Sea $g(x) = f''(x)$ entonces $g'(x) = f'''(x)$ y el esquema iterativo anterior, toma la forma*

$$\left\{ \begin{array}{r} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f''(x_n)]^2 f^2(x_n)}{2 [f'(x_n)]^2 (f'(x_n) f''(x_n) + f''(x_n) f(x_n))} \end{array} \right. \tag{164}$$

Algoritmo 126 *Sea $g(x) = \dfrac{1}{f'(x)}$ entonces $g'(x) = g'(x) = -\dfrac{f''(x)}{[f'(x)]^2}$ y el esquema iterativo anterior, toma la forma*

$$\left\{ \begin{array}{r} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n) f'(x_n)}{2 [f'(x_n)]^4 - f''(x_n) f(x_n)} \end{array} \right. \tag{165}$$

Algoritmo 127 *Sea* $g(x) = \dfrac{f''(x)}{f'(x)}$ *entonces* $g'(x) = \dfrac{f'''(x)\, f'(x) - [f''(x)]^2}{[f'(x)]^2}$ *y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n)\, f^2(x_n)}{[f'(x_n)]^3\, f''(x_n) + f'''(x_n)\, f'(x_n)\, f^2(x_n) - [f''(x_n)]^2\, f^2(x_n)} \end{cases} \tag{166}$$

Algoritmo 128 *Sea* $g(x) = \dfrac{1}{2}\dfrac{f''(x)}{f'(x)}$ *entonces* $g'(x) = \dfrac{1}{2}\dfrac{f'''(x)\, f'(x) - [f''(x)]^2}{[f'(x)]^2}$ *y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n)\, f^2(x_n)}{[f'(x_n)]^3 \left(2f''(x_n) + f'''(x_n)\, f'(x_n) - [f''(x_n)]^2\right)} \end{cases} \tag{167}$$

Ahora vamos a considerar la familia de funciones exponenciales:

Algoritmo 129 *Sea* $g(x) = e^{-\alpha x}$ *entonces* $g'(x) = -\alpha e^{-ax}$ *y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n)\, f^2(x_n)}{[f'(x_n)]^3\, f''(x_n) + f'''(x_n)\, f'(x_n)\, f^2(x_n) - [f''(x_n)]^2\, f^2(x_n)} \end{cases} \tag{168}$$

Algoritmo 130 *Sea* $g(x) = e^{-\alpha f(x)}$ *entonces* $g'(x) = -\alpha f'(x)\, e^{-af(x)}$ *y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n)\, f^2(x_n)}{2\,[f'(x_n)]^2\,[1 - \alpha f(x_n)]} \end{cases} \tag{169}$$

Generalizando el esquema anterior,

Algoritmo 131 *Sea* $g(x) = f(x)e^{-\alpha x}$ *entonces* $g'(x) = f'(x)e^{-\alpha x} - \alpha e^{-ax}f(x)$ *y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{4[f'(x_n)]^2 - 2\alpha f'(x_n) f(x_n)} \end{cases} \tag{170}$$

Algoritmo 132 *Sea* $g(x) = f(x)e^{-\alpha f(x)}$ *entonces* $g'(x) = f'(x)e^{-\alpha f(x)} - \alpha f'(x)e^{-af(x)}f(x)$ *y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{4[f'(x_n)]^3 - 2\alpha [f'(x_n)]^2 f(x_n)} \end{cases} \tag{171}$$

Algoritmo 133 *Sea* $g(x) = e^{\alpha x}$ *entonces* $g'(x) = \alpha e^{ax}$ *y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{2[f'(x_n)]^2 (f'(x_n) + \alpha f(x_n))} \end{cases} \tag{172}$$

Algoritmo 134 *Sea* $g(x) = e^{\alpha f(x)}$ *entonces* $g'(x) = \alpha f'(x)e^{af(x)}$ *y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{2[f'(x_n)]^2 (f'(x_n) + \alpha f(x_n) f'(x_n))} \end{cases} \tag{173}$$

Algoritmo 135 *Sea* $g(x) = f(x)e^{\alpha x}$ *entonces* $g'(x) = f'(x)e^{\alpha x} + \alpha e^{ax}f(x)$ *y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{2[f'(x_n)]^2 (2f'(x_n) + \alpha f(x_n))} \end{cases} \tag{174}$$

Notemos que si $\alpha = 0$, el algoritmo anterior toma la forma

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{4\left[f'(x_n)\right]^3} \end{cases} \tag{175}$$

Algoritmo 136 *Sea $g(x) = e^{\frac{\alpha}{f(x)}}$ entonces $g'(x) = -\alpha \dfrac{f'(x)}{f^2(x)} e^{\frac{\alpha}{f(x)}}$ y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^3(x_n)}{2\left[f'(x_n)\right]^2 \left(f(x_n) f'(x_n) - \alpha f'(x_n)\right)} \end{cases} \tag{176}$$

Notemos que si $\alpha = 0$, el algoritmo anterior toma la forma

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{2\left[f'(x_n)\right]^3} \end{cases} \tag{177}$$

Algoritmo 137 *Sea $g(x) = f(y)$ entonces $g'(x) = f'(y) \dfrac{f''(x) f(x)}{\left[f'(x)\right]^2}$ y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n) f(y)}{2\left[f'(x_n)\right]^3 f(y) - 2 f'(y) f''(x_n) f^2(x_n)} \end{cases} \tag{178}$$

Algoritmo 138 *Sea $g(x) = \dfrac{1}{f(y)}$ entonces $g'(x) = -\dfrac{f'(y)}{\left[f(y)\right]^2} \dfrac{f''(x) f(x)}{\left[f'(x)\right]^2}$ y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n) f(y) \left[f'(x_n)\right]^2}{2\left[f'(x_n)\right]^2 \left(\left[f'(x_n)\right]^3 f(y) - f'(y) f''(x_n) f^2(x_n)\right)} \end{cases} \tag{179}$$

Algoritmo 139 *Sea* $g(x) = \dfrac{1}{f'(x)}$ *entonces* $g'(x) = -\dfrac{f''(x)}{[f'(x)]^2}$ *y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n) f'(x_n)}{2[f'(x_n)]^2 \left([f'(x_n)]^2 - f(x_n) f''(x_n)\right)} \end{cases} \tag{180}$$

Algoritmo 140 *Sea* $g(x) = \dfrac{1}{f''(x)}$ *entonces* $g'(x) = -\dfrac{f'''(x)}{[f''(x)]^2}$ *y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f''(x_n)]^2 f^2(x_n) f(x_n)}{2[f'(x_n)]^2 [f''(x_n) f'(x_n) - f'''(x_n) f(x_n)]} \end{cases} \tag{181}$$

Algoritmo 141 *Sea* $g(x) = \dfrac{f''(x)}{f'(x)}$ *entonces* $g'(x) = \dfrac{f'''(x) f'(x) - [f''(x)]^2}{[f'(x)]^2}$ *y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f''(x_n)]^2 f^2(x_n) f'(x_n)}{2[f'(x_n)]^2 \left[f''(x_n)[f'(x_n)]^2 + f(x_n) f'(x_n) f'''(x_n) - [f''(x_n)]^2 f(x_n)\right]} \end{cases} \tag{182}$$

Algoritmo 142 *Sea* $g(x) = e^{\frac{f(x)}{f'(x)}}$ *entonces* $g'(x) = \left(1 - \dfrac{[f''(x)]^2 f(x)}{[f'(x)]^2}\right) e^{\frac{f(x)}{f'(x)}}$ *y el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{2\left[[f'(x_n)]^3 + [f'(x_n)]^2 f(x_n) - [f''(x_n)]^2 f^2(x_n)\right]} \end{cases} \tag{183}$$

Analicemos el caso cuando $g(x) = \dfrac{1}{\phi(x)}$ entonces $g'(x) = -\dfrac{\phi'(x)}{[\phi(x)]^2}$ y el esquema iterativo anterior, toma la forma

$$
\begin{aligned}
y_n &= x_n - \frac{f(x_n)}{f'(x_n)} \\
H(x_n) &= y_n - \frac{\dfrac{[f(x_n)]^2 f''(x_n)}{\phi(x_n)}}{2[f'(x_n)]^2\left(\dfrac{f'(x_n)}{\phi(x_n)} - \dfrac{\phi'(x_n) f(x_n)}{[\phi(x_n)]^2}\right)}
\end{aligned} \tag{184}
$$

sustituyendo $\phi(x) = x - \dfrac{f(x)}{f'(x)}$ y $\phi'(x) = \dfrac{f'(x) f''(x)}{(f'(x))^2}$, el esquema anterior, se convierte en el siguiente

Algoritmo 143

$$
\begin{cases}
y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\
\\
x_{n+1} = y_n - \dfrac{[f(x_n)]^2 f''(x_n)[f'(x_n) - f(x_n)]}{2[f'(x_n)]^2\left([f'(x_n)]^3 - [f'(x_n)]^2 - f^2(x_n) f''(x_n)\right)}
\end{cases} \tag{185}
$$

Algoritmo 144 *Sea* $g(x) = \dfrac{1}{2}\dfrac{f''(x)}{f'(x)}$ *entonces* $g'(x) = \dfrac{1}{2}\dfrac{f'''(x) f'(x) - [f''(x)]^2}{[f'(x)]^2}$ *y el esquema iterativo anterior, toma la forma*

$$
\begin{cases}
y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\
\\
x_{n+1} = y_n - \dfrac{[f''(x_n)]^2 f^2(x_n) f'(x_n)}{[f'(x_n)]^2\left[2f''(x_n)[f'(x_n)]^2 + f(x_n) f'(x_n) f'''(x_n) - [f''(x_n)]^2 f(x_n)\right]}
\end{cases} \tag{186}
$$

Algoritmo 145 *Sea* $g(x) = y$, *donde* $y = x - \dfrac{f(x)}{f'(x)}$, *entonces* $g'(x) = \dfrac{f''(x) f(x)}{[f'(x)]^2}$ *y el esquema iterativo anterior, toma la forma*

$$
\begin{cases}
y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\
\\
x_{n+1} = y_n - \dfrac{[f'(x_n)]^2 f^2(x_n) f''(x_n)(y_n)}{2[f'(x_n)]^2\left[[f'(x_n)]^3(y_n) + f^2(x_n) f''(x_n)\right]}
\end{cases} \tag{187}
$$

Algoritmo 146 *Sea* $y = x - \dfrac{f(x)}{f'(x)}$ *y* $g(x) = e^{-\alpha y}$ *entonces* $g'(x) = -\alpha y' e^{-ay}$, *de donde*

$$
\begin{cases}
y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\
\\
x_{n+1} = y_n - \dfrac{f''(x) f^2(x_n)}{2\left[[f'(x)]^3 - \alpha f^2(x) f''(x)\right]}
\end{cases} \tag{188}
$$

Algoritmo 147 *Sea* $y = x - \dfrac{f(x)}{f'(x)}$ *y* $g(x) = e^{-\alpha f(y)}$ *entonces* $g'(x) = -\alpha f'(y)\, y' e^{-af(y)}$, *luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x)\, f^2(x_n)}{2\left[[f'(x)]^3 - \alpha f'(y_n)\, f^2(x)\, f''(x)\right]} \end{cases} \tag{189}$$

Algoritmo 148 *Sea* $g(x) = ye^{-\alpha y}$ *entonces* $g'(x) = y'e^{-\alpha x} - \alpha y' e^{-ay}$, *entonces*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{yf''(x)\, f^2(x_n)}{2\left[y\,[f'(x)]^3 + f^2(x)\, f''(x_n) - \alpha f^2(x)\, f''(x_n)\right]} \end{cases} \tag{190}$$

Algoritmo 149 *Si* $\alpha = 0$, *obtenemos el metodo iterativo*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{yf''(x)\, f^2(x_n)}{2\left[y\,[f'(x)]^3 + f^2(x)\, f''(x_n)\right]} \end{cases} \tag{191}$$

Algoritmo 150 *Si* $\alpha = 1$, *se tiene el método iterativo*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x)\, f^2(x_n)}{2\,[f'(x)]^3} \end{cases} \tag{192}$$

Algoritmo 151 *Sea* $g(x) = f(x)\, e^{-\alpha y}$ *entonces* $g'(x) = [f'(x) - \alpha y' f(x)]\, e^{-ay}$, *luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x)\, f^2(x_n)}{2\left[2\,[f''(x)]^3 - \alpha f^2(x)\, f''(x)\right]} \end{cases} \tag{193}$$

Algoritmo 152 *Si* $\alpha = 0$, *se obtiene el método*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x)\, f^2(x_n)}{4\,[f''(x)]^3} \end{cases} \tag{194}$$

Algoritmo 153 *Sea* $g(x) = e^{\alpha y}$ *entonces* $g'(x) = \alpha y' e^{ax}$, *en consecuencia*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x) f^2(x_n)}{2\left[[f'(x)]^3 + \alpha f^2(x) f''(x)\right]} \end{cases} \tag{195}$$

Algoritmo 154 *Sea* $g(x) = e^{\alpha f(y)}$ *entonces* $g'(x) = \alpha f'(y) e^{af(y)}$, *luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n) [f'(x_n)]^2}{2\left[[f'(x_n)]^5 + \alpha f'(y_n) f^2(x_n) [f''(x_n)]^2 f(x)\right]} \end{cases} \tag{196}$$

Si $\alpha = 0$ el esquema iterativo anterior, toma la forma

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x) f^2(x_n)}{2[f'(x)]^3} \end{cases}$$

Algoritmo 155 *Sea* $g(x) = e^{-\frac{\alpha}{y}}$ *entonces* $g'(x) = -\alpha \dfrac{[f'(x)]^2}{f(x) f''(x)} e^{-\frac{\alpha}{y}}$, *luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f''(x)]^2 f^2(x_n)}{2[f'(x)]^2 \left(f'(x) f''(x) - \alpha [f'(x)]^2\right)} \end{cases} \tag{197}$$

Algoritmo 156 *Si* $\alpha = 0$ *el esquema iterativo anterior, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f''(x)]^2 f^2(x_n)}{2[f'(x)]^3 f''(x)} \end{cases} \tag{198}$$

Algoritmo 157 *Sea* $g(x) = e^{-\frac{\alpha}{f(y)}}$ *entonces* $g'(x) = -\alpha \dfrac{1}{f'(y)} e^{-\frac{\alpha}{f(y)}}$, *en consecuencia*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f''(x)]^2 f^2(x_n) f'(y_n)}{2[f'(x)]^2 \left(f'(x) - \alpha [f'(x)]^2\right)} \end{cases} \tag{199}$$

Algoritmo 158 *Si $\alpha = 0$, el esquema iterativo toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f''(x)]^2 f^2(x_n) f'(y_n)}{2[f'(x)]^3} \end{cases} \tag{200}$$

Algoritmo 159 *Sea $g(x) = e^{-\alpha \frac{f(y)}{f'(y)}}$ entonces $g'(x) = -\alpha \left[\dfrac{f(y)}{f'(y)}\right]' e^{-\frac{f(y)}{f'(y)}}$ y obtenemos*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x) f^2(x_n) [f'(y_n)]^2}{2[f'(x)]^2 \left(f'(x)[f'(y_n)]^2 - \alpha [f'(y)]^2 f(x) + \alpha f''(y_n) f(y) f(x_n)\right)} \end{cases} \tag{201}$$

Si $\alpha = 0$, el esquema iterativo toma la forma

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x) f^2(x_n)}{2[f'(x)]^3} \end{cases}$$

Algoritmo 160 *Sea $g(x) = e^{-\alpha \frac{f'(y)}{f''(y)}}$ entonces $g'(x) = -\alpha \left[\dfrac{f'(y)}{f''(y)}\right]' e^{-\alpha \frac{f'(y)}{f''(y)}}$, en consecuencia*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x) f^2(x_n) [f''(y_n)]^2}{2[f'(x)]^2 \left(f'(x)[f''(y_n)]^2 - \alpha [f''(y)]^2 f(x) + \alpha f'''(y_n) f'(y) f(x_n)\right)} \end{cases} \tag{202}$$

Si $\alpha = 0$, el esquema iterativo toma la forma

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{2[f'(x_n)]^3} \end{cases}$$

Algoritmo 161 *Sea $g(x) = f(y) e^{-\alpha x}$ entonces $g'(x) = e^{-\alpha x}[f'(y) - \alpha f(y)]$, luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n) f(y_n)}{2[f'(x_n)]^2 (f'(x_n) f(y_n) + \alpha f'(y_n) f(x_n) - \alpha f(x_n) f(y_n))} \end{cases} \tag{203}$$

Si $\alpha = 0$, el esquema iterativo toma la forma

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{2[f'(x_n)]^3} \end{cases}$$

Algoritmo 162 *Sea* $g(x) = f(x) e^{-\alpha f(y)}$ *entonces* $g'(x) = e^{-\alpha f(y)} [f'(x) - \alpha f'(y) f(x)]$ *, en consecuencia*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{2[f'(x_n)]^2 (2f'(x_n) - \alpha f(x_n) f'(y_n))} \end{cases} \tag{204}$$

Si $\alpha = 0$, el esquema iterativo toma la forma

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{4[f'(x_n)]^3} \end{cases}$$

Algoritmo 163 *Sea* $g(x) = f'(x) e^{-\alpha y}$ *entonces* $g'(x) = e^{-\alpha y} [f''(x) - \alpha y' f'(x)]$ *y obtenemos*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{2\left[[f'(x_n)]^3 + f''(y_n) f'(x_n) f(x_n) - \alpha f(x_n) f''(x_n)\right]} \end{cases} \tag{205}$$

Algoritmo 164 *Si* $\alpha = 0$, *el esquema iterativo toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f''(x_n) f^2(x_n)}{2\left[[f'(x_n)]^3 + f''(y_n) f'(x_n) f(x_n)\right]} \end{cases} \tag{206}$$

9.6. Caso VI : $H(x) = \phi(x) + \lambda(f(\phi(x))\, g(\phi(x)))$

Ahora vamos a generalizar otro esquema iterativo, considerando la forma funcional definida por

$$H_2(x) = \phi(x) + \lambda(f(\phi(x))\, g(\phi(x)))$$

donde

$\phi(x)$: Función iterativa predictora de orden $p \geq 1$

$g(x)$: Función arbitraria

λ : Multiplicador de Lagrange

Por otro lado, la condición de optimalidad implica que

$$\lambda = -\frac{1}{[f'(\phi(x))\, g(\phi(x)) + g'(\phi(x))\, f'(\phi(x))]}$$

entonces el esquema iterativo principal se convierte en

$$H(x) = \phi(x) - \frac{f'(\phi(x))\, g(\phi(x))}{(f'(\phi(x))\, g(\phi(x)) + g'(\phi(x))\, f(\phi(x)))}$$

Es meritorio señalar que el nuevo esquema obtenido incluye al reconocido método de Noor [65] como un caso particular. También notemos que para $\phi(x) = x$ obtenemos

$$H(x) = x - \frac{f(x)\, g(x)}{f'(x)\, g(x) + g'(x)\, f(x)}$$

fórmula que fue obtenida en el algoritmo dedicado al método de Newton.

Igual que en los casos anteriores, consideremos la función iterativa definida por el método de Newton

$$\phi(x) = y = x - \frac{f(x)}{f'(x)}$$

y de esta manera, obtenemos el siguiente esquema iterativo

$$\left\{ \begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ H(x_n) = y_n - \dfrac{f(y_n)\, g(y_n)}{(f'(y_n)\, g(y_n) + g'(y_n)\, f(y_n))} \end{array} \right. \tag{207}$$

notemos que si dividimos la fracción del esquema anterior por $g(y_n)$, y si reemplazamos la expresón $\dfrac{g'(y_n)}{g(y_n)}$ por $\dfrac{g'(x_n)}{g(x_n)}$, obtendremos un nuevo método iterativo equivalente al anterior, el cual está definido por

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ H(x_n) = y_n - \dfrac{f(y_n)\, g(x_n)}{(f'(y_n)\, g(x_n) + g'(x_n)\, f(y_n))} \end{cases} \tag{208}$$

el cual resulta más sencillo de manipular y al tomar la función auxiliar diversas formas funcionales, obtenemos una nueva familia de métodos iterativos, tales son los casos de las funciones exponenciales, donde dichos esquemas generan múltiples formas.

De las relaciones

$$\begin{aligned} f''(x) &= \frac{f'(y) - f'(x)}{y - x} \\ f''(x) &= \frac{2f(y)\,[f'(x)]^2}{[f(x)]^2} \end{aligned}$$

obtenemos la expresión

$$f'(y) \approx \frac{f'(x)}{f(x)}\,[f(x) - 2f(y)]$$

y sustituyéndola en el esquema iterativo

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ H(x_n) = y_n - \dfrac{f(y_n)\, g(y_n)}{(f'(y_n)\, g(y_n) + g'(y_n)\, f(y_n))} \end{cases}$$

se tiene un nuevo método iterativo definido por

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ H(x_n) = y_n - \dfrac{f(x_n)\, f(y_n)\, g(x_n)}{f'(y_n)\,[f(x_n) - 2f(y_n)] + f(x_n)\, f(y_n)\, g'(x_n)} \end{cases} \tag{209}$$

Ahora vamos a continuar profundizando con el hallazgo de los principales esquemas iterativos encontrados y a partir de aquí, vamos a considerar diversas variantes de la función auxiliar $g(x)$ para poder determinar nuevos métodos iterativos.

Casos Especiales:

Caso I:

Sea el método de Newton expresado por

$$x_{k+1} = x_k - \frac{h(x)}{h'(x)}$$

Si $h(x) = \dfrac{f(x)}{g(x)}$, entonces $h'(x) = \dfrac{f'(x)\,g(x) - g'(x)\,f(x)}{[g(x)]^2}$, entonces el esquema iterativo toma la forma

$$x_{k+1} = x_k - \frac{f(x)\,g(x)}{f'(x)\,g(x) - g'(x)\,f(x)}$$

Si la función auxiliar $g(x) = c$, entonces obtenemos

$$x_{k+1} = x_k - \frac{f(x)}{f'(x)}$$

el cual constituye el método clásico de Newton.

Si la función auxiliar $g(x) = f(x)$, entonces el esquema iterativo colapsa.

Algoritmo 165 *Si la función auxiliar $g(x) = f'(x)$, obtenemos*

$$x_{k+1} = x_k - \frac{f(x_n)\,f'(x_n)}{[f'(x_n)]^2 - f(x_n)\,f''(x_n)} \tag{210}$$

Algoritmo 166 *Si la función auxiliar $g(x) = f''(x)$, luego*

$$x_{k+1} = x_k - \frac{f(x_n)\,f''(x_n)}{f'(x_n)\,f''(x_n) - f'''(x_n)\,f(x_n)} \tag{211}$$

Algoritmo 167 *Sea $y = x - \dfrac{f(x)}{f'(x)}$, consideremos la función auxiliar $g(x) = f(y)$, entonces*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n)\,f(y_n)}{f'(x_n)\,f(y_n) - f'(y_n)\,f(x_n)} \end{cases} \tag{212}$$

Ahora, considerando la relaciónn fundamental

$$x_{n+1} = \phi(x_n) - \frac{f(\phi(x_n))\,g(\phi(x_n))}{f'(\phi(x_n))\,g(\phi(x_n)) + g'(\phi(x_n))\,f(\phi(x_n))}$$

y haciendo

$$\phi(x_n) = y_n = x_n - \frac{f(x_n)}{f'(x_n)}$$

Algoritmo 168 *Sea* $g(y) = 1,$ *entonces el método iterativo es*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(y_n)} \end{cases} \tag{213}$$

Algoritmo 169 *Sea* $g(y) = y,$ *luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{yf'(y_n) + f(y_n)} \end{cases} \tag{214}$$

Algoritmo 170 *Sea la función auxiliar* $g(y) = f(y),$ *entonces* $g'(y) = f'(y),$ *en consecuencia*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{2f'(y_n)} \end{cases} \tag{215}$$

Algoritmo 171 *Sea* $g(x) = f'(y),$ *entonces* $g'(y) = f''(y)$ *, luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n) f'(y_n)}{[f'(y_n)]^2 + f''(y_n) f(y_n)} \end{cases} \tag{216}$$

Algoritmo 172 *Sea* $g(x) = f''(y),$ *entonces* $g'(y) = f'''(y),$ *obtenemos*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n) f''(y_n)}{f'(y_n) f''(y_n) + f'''(y_n) f(y_n)} \end{cases} \tag{217}$$

Algoritmo 173 *Sea* $g(y) = \dfrac{f(y)}{f'(y)}$, *entonces* $g'(y) = \dfrac{[f'(y)]^2 - f''(y)\,f(y)}{[f'(y)]^2}$ *y el método toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{[f(y_n)]^2}{f(y_n)\,f'(y_n) + f'(y_n) - f(y_n)} \end{cases} \tag{218}$$

Algoritmo 174 *Sea* $g(y) = \dfrac{f''(y)}{f'(y)}$, *entonces* $g'(y) = \dfrac{f'(y)\,f'''(y) - [f''(y)]^2}{[f'(y)]^2}$ *y se obtiene*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)\,f'(y_n)\,f''(y_n)}{[f'(y_n)]^2\,f''(y_n) + f'(y_n)\,f'''(y_n) - [f''(y_n)]^2} \end{cases} \tag{219}$$

Algoritmo 175 *Sea* $g(y) = e^{-\alpha f(y)}$ *entonces* $g'(y) = -\alpha f'(y)\,e^{\alpha f(y)}$, *en consecuencia*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(y_n) - \alpha f'(y_n)\,f(y_n)} \end{cases} \tag{220}$$

Algoritmo 176 *Sea* $g(y) = f(y)\,e^{-\alpha f(y)}$ *entonces* $g'(y) = f'(y)\,e^{-\alpha f(y)} - \alpha f'(y)\,e^{\alpha f(y)} f(y)$, *luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^2(y_n)}{f'(y_n)\,f(y_n) + f'(y_n) - \alpha f'(y_n)\,f^2(y_n)} \end{cases} \tag{221}$$

Algoritmo 177 *Si* $\alpha = 0$, *se obtiene el metodo iterativo es*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^2(y_n)}{f'(y_n)\,f(y_n) + f'(y_n)} \end{cases} \tag{222}$$

Algoritmo 178 *Sea* $g(y) = e^{-\alpha f'(y)}$ *entonces* $g'(y) = -\alpha f''(y)\,e^{-\alpha f'(y)}$, *luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(y_n) - \alpha f''(y_n)\,f(y_n)} \end{cases} \tag{223}$$

Algoritmo 179 *Sea* $g(y) = e^{-\alpha f''(y)}$ *entonces* $g'(y) = -\alpha f'''(y)\, e^{-\alpha f''(y)}$, *en consecuencia*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(y_n) - \alpha f'''(y_n)\, f(y_n)} \end{cases} \tag{224}$$

Algoritmo 180 *Sea* $g(y) = f^2(y)$ *entonces* $g'(y) = 2f(y)\, f'(y)$ *, luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^3(y_n)}{3f'(y_n)\, f^2(y_n)} \end{cases} \tag{225}$$

Algoritmo 181 *Sea* $g(y) = \dfrac{1}{f(y)}$ *entonces* $g'(y) = -\dfrac{f'(y)}{f^2(y)}$ *, y el esquema iterativo toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n + \dfrac{f(y_n)}{f'(y_n)} \end{cases} \tag{226}$$

Algoritmo 182 *Sea* $g(y) = \dfrac{f(y)}{f'(y)}$ *entonces* $g'(y) = \dfrac{[f'(y)]^2 - f''(y)\, f(y)}{[f'(y)]^2}$ *, luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^2(y_n)\, f'(y_n)}{[f'(y_n)]^3 + [f'(y_n)]^2 f(y_n) - f''(y_n)\, f^2(y_n)} \end{cases} \tag{227}$$

Algoritmo 183 *Sea* $g(y) = \dfrac{f''(y)}{f'(y)}$ *entonces* $g'(y) = \dfrac{f'''(y)\, f'(y) - [f''(y)]^2}{[f'(y)]^2}$ *, y se obtiene*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)\, f'(y_n)\, f''(y_n)}{[f'(y_n)]^2 f''(y_n) + f'''(y_n)\, f'(y_n)\, f(y_n) - f(y_n)\, [f'(y_n)]^2} \end{cases} \tag{228}$$

Algoritmo 184 *Sea* $g(y) = e^{-\frac{\alpha}{f(y)}}$ *entonces* $g'(y) = -\alpha \dfrac{f'(y)}{f^2(y)} e^{-\frac{\alpha}{f(y)}}$ *, en consecuencia*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^2(y_n)}{f'(y_n)\, f(y_n) - \alpha f'(y_n)} \end{cases} \tag{229}$$

Si $\alpha = 0$ el esquema iterativo toma la forma

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(y_n)} \end{cases}$$

Algoritmo 185 *Sea* $g(y) = e^{-\frac{\alpha}{f'(y)}}$ *entonces* $g'(y) = -\alpha \dfrac{f''(y)}{[f'(y)]^2} e^{-\frac{\alpha}{f'(y)}}$ *, y el esquema iterativo toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)[f'(y_n)]^2}{[f'(y_n)]^3 - \alpha f(y_n) f''(y_n)} \end{cases} \tag{230}$$

Si $\alpha = 0$, el esquema iterativo toma la forma

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)[f'(y_n)]^2}{[f'(y_n)]^3} \end{cases}$$

Algoritmo 186 *Sea* $g(y) = e^{-\frac{\alpha}{f''(y)}}$ *entonces* $g'(y) = -\alpha \dfrac{f'''(y)}{[f''(y)]^2} e^{-\frac{\alpha}{f''(y)}}$ *, luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)[f''(y_n)]^2}{f'(y_n)[f''(y_n)]^2 - \alpha f(y_n) f'''(y_n)} \end{cases} \tag{231}$$

Si $\alpha = 0$, el esquema iterativo toma la forma

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)[f''(y_n)]^2}{f'(y_n)[f''(y_n)]^2} \end{cases}$$

Algoritmo 187 *Sea* $g(y) = e^{-\frac{\alpha f''(y)}{f'(y)}}$ *entonces* $g'(y) = -\alpha \dfrac{f'''(y) f'(y) - [f''(y)]^2}{[f'(y)]^2} e^{-\frac{\alpha f''(y)}{f'(y)}}$ *, en consecuencia*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)[f'(y_n)]^2}{[f'(y_n)]^3 - \alpha f'''(y_n) f'(y_n) f(y_n) + \alpha f(y_n)[f''(y_n)]^2} \end{cases} \tag{232}$$

Algoritmo 188 *Sea* $g(y) = e^{\alpha f(y)}$ *entonces* $g'(y) = \alpha f'(y) e^{\alpha f(y)}$, *luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(y_n) + \alpha f'(y_n) f(y_n)} \end{cases} \tag{233}$$

Algoritmo 189 *Sea* $g(y) = f(y) e^{\alpha f(y)}$ *entonces* $g'(y) = f'(y) e^{-\alpha f(y)} + \alpha f'(y) e^{\alpha f(y)} f(y)$, *y el esquema iterativo toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{2f'(y_n) + \alpha f'(y_n) f(y_n)} \end{cases} \tag{234}$$

De la relacion fundamental

$$x_{n+1} = y_n - \frac{f(x_n) f(y_n) g(x_n)}{f'(x_n)[f(x_n) - 2f(y_n)] g(x_n) + f(x_n) f(y_n) g'(x_n)}$$

Algoritmo 190 *Sea* $g(x) = f(y)$, *entonces* $g'(x) = f'(y)$ *y el esquema iterativo, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n)[f(y_n)]^2}{f'(x_n)[f(x_n) - 2f(y_n)] f(y_n) + f(x_n) f(y_n) f'(y_n)} \end{cases} \tag{235}$$

Algoritmo 191 *Sea* $g(x) = f'(y)$, *entonces* $g'(x) = f''(y)$ *y el esquema iterativo es*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n) f(y_n) f'(y_n)}{f'(x_n)[f(x_n) - 2f(y_n)] f'(y_n) + f(x_n) f(y_n) f''(y_n)} \end{cases} \tag{236}$$

Algoritmo 192 *Sea* $g(x) = f''(y)$, *entonces* $g'(x) = f'''(y)$ *y el esquema toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n) f(y_n) f''(y_n)}{f'(x_n)[f(x_n) - 2f(y_n)] f''(y_n) + f(x_n) f(y_n) f'''(y_n)} \end{cases} \tag{237}$$

De forma general, si la función auxiliar $g(x) = f^{(n)}(y)$, entonces $g'(x) = f^{(n+1)}(y)$ y en consecuencia

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)} \\ x_{k+1} &= y_k - \frac{f(x_n) f(y_n) f^{(n)}(y_n)}{f'(x_n)[f(x_n) - 2f(y_n)] f^{(n)}(y_n) + f(x_n) f(y_n) f^{(n+1)}(y_n)} \end{aligned}$$

Algoritmo 193 *Sea* $g(x) = \dfrac{1}{f'(y)}$, *entonces* $g'(x) = -\dfrac{f''(y)}{[f'(y)]^2}$ *y el método es*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n) f(y_n) f'(y_n)}{f'(x_n) f'(y_n)[f(x_n) - 2f(y_n)] - f(x_n) f(y_n) f''(y_n)} \end{cases} \tag{238}$$

Algoritmo 194 *Sea* $g(x) = \dfrac{1}{f(y)}$, *entonces* $g'(x) = -\dfrac{f'(y)}{[f(y)]^2}$ *, luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n) f(y_n)}{f'(x_n)[f(x_n) - 2f(y_n)] + f(x_n) f'(y_n)} \end{cases} \tag{239}$$

Algoritmo 195 *Sea* $g(x) = \dfrac{f(y)}{f'(y)}$, *entonces* $g'(x) = -\dfrac{[f'(y)]^2 - f(y) f'(y)}{[f'(y)]^2}$ *y en consecuencia*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n) f'(x_n) f(y_n)}{f'(x_n)[f(x_n) - 2f(y_n)] + f(y_n)\left[[f'(x_n)]^2 - f'(x_n) f''(x_n)\right]} \end{cases} \tag{240}$$

Algoritmo 196 *Sea* $g(x) = \dfrac{f'(y)}{f(y)}$, *entonces* $g'(x) = -\dfrac{f''(y) f(y) - [f'(y)]^2}{[f(y)]^2}$ *y el método iterativo es*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n) f'(x_n) f(y_n)}{f'(x_n)[f(x_n) - 2f(y_n)] + f(y_n)\left[f''(x_n) f'(x_n) - [f'(x_n)]^2\right]} \end{cases} \tag{241}$$

Algoritmo 197 *Sea* $g(x) = f^2(y)$, *entonces* $g'(x) = 2f(y) f'(y)$ *y obtenemos*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n) f(y_n)}{f'(x_n)[f(x_n) - 2f(y_n)] + 2f(x_n) f'(y_n)} \end{cases} \tag{242}$$

Algoritmo 198 *Sea* $g(x) = \dfrac{1}{f^2(y)}$, *entonces* $g'(x) = -\dfrac{2f'(y)}{f^3(y)}$, *luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n) f(y_n)}{f(x_n)[f(x_n) - 2f(y_n)] - 2f(x_n) f'(y_n)} \end{cases} \tag{243}$$

Algoritmo 199 *Sea* $g(x) = \dfrac{f''(y)}{f'(y)}$, *entonces* $g'(x) = \dfrac{f'''(y) f'(y) - [f''(y)]^2}{[f'(y)]^2}$ *y obtenemos*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n) f'(x_n) f''(x_n) f(y_n)}{[f'(x_n)]^2 [f(x_n) - 2f(y_n)] f''(x_n) + f(x_n) f(y_n) \left[f'(x_n) f'''(x_n) - [f''(x_n)]^2\right]} \end{cases} \tag{244}$$

Caso 2:

Consideremos la variante del método de Newton

$$x_{n+1} = y_n - \frac{f(y_n) g(x_n)}{f'(x_n) g(x_n) + g'(x_n) f(x_n)}$$

donde

$$y_n = x_n - \frac{f(x_n)}{f'(x_n)}$$

Algoritmo 200 *Sea* $g(x) = \dfrac{f'(x)}{f''(x)}$, *entonces* $g'(x) = \dfrac{[f''(x)]^2 - f'''(x) f'(x)}{[f''(x)]^2}$, *y el esquema iterativo, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n) f'(x_n) f''(x_n)}{f''(x_n)[f'(x_n)]^2 + [f''(x_n)]^2 f(x_n) - f'''(x_n) f'(x_n) f(x_n)} \end{cases} \tag{245}$$

Algoritmo 201 *Sea* $g(x) = \dfrac{f''(x)}{f'(x)}$, *entonces* $g'(x) = \dfrac{f'''(x) f'(x) - [f''(x)]^2}{[f'(x)]^2}$, *y el esquema iterativo, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n) f'(x_n) f''(x_n)}{f''(x_n)[f'(x_n)]^2 + f'''(x_n) f'(x_n) f(x_n) - [f''(x_n)]^2 f(x_n)} \end{cases} \tag{246}$$

Algoritmo 202 *Sea* $g(x_n) = e^{\alpha x}$, *entonces* $g'(x_n) = \alpha e^{\alpha x}$ *y el esquema iterativo, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n) + \alpha f(x_n)} \end{cases} \tag{247}$$

Algoritmo 203 *Sea* $g(x) = e^{-\alpha x}$, *entonces* $g'(x_n) = -\alpha e^{\alpha x}$, *luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n) - \alpha f(x_n)} \end{cases} \tag{248}$$

Algoritmo 204 *Sea* $g(x) = e^{\alpha f(x)}$, *entonces* $g'(x) = \alpha f'(x) e^{\alpha f(x)}$, *luego*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n) - \alpha f'(x_n) f(x_n)} \end{cases} \tag{249}$$

Algoritmo 205 *Sea* $g(x_n) = e^{\frac{\alpha}{f(x)}}$, *entonces* $g'(x) = -\alpha \dfrac{f'(x)}{f^2(x)} e^{\frac{\alpha}{f(x)}}$, *y en consecuencia*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n) f(x_n)}{f'(x_n) f(x_n) - \alpha f'(x_n)} \end{cases} \tag{250}$$

Algoritmo 206 *Si la función auxiliar es* $g(x) = e^{\frac{f(x)}{f'(x)}}$, *entonces* $g'(x) = \dfrac{[f'(x)]^2 - f''(x) f(x)}{[f'(x)]^2} e^{\frac{f(x)}{f'(x)}}$ *, y el esquema iterativo, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)[f'(x_n)]^2}{[f'(x_n)]^3 + [f'(x_n)]^2 f(x_n) - f''(x_n) f^2(x_n)} \end{cases} \tag{251}$$

Algoritmo 207 *Si la función auxiliar es* $g(x) = 1$, *entonces* $g'(x) = 0$ *, y el esquema iterativo, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(x_n)} \end{cases} \tag{252}$$

Algoritmo 208 *Si la función auxiliar es* $g(x) = \dfrac{1}{f(y)}$, *entonces* $g'(x) = -\dfrac{f'(y)}{f^2(y)}$ *, y el esquema iterativo, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^2(y_n)}{f'(x_n) f(y_n) - f'(y_n) f(x_n)} \end{cases} \tag{253}$$

Algoritmo 209 *Si la función auxiliar es* $g(x) = f(y)$, *entonces* $g'(x) = f'(y)$ *, y el esquema iterativo, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^2(y)}{f'(x_n) f(y) + f'(y) f(x_n)} \end{cases} \tag{254}$$

Algoritmo 210 *Si la función auxiliar es* $g(x) = f'(y)$, *entonces* $g'(x) = f''(y)$ *, y el esquema iterativo, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y) f'(y_n)}{f'(x_n) f'(y_n) + f''(y_n) f(x_n)} \end{cases} \tag{255}$$

Algoritmo 211 *Si la función auxiliar es $g(x) = f''(y)$, entonces $g'(x) = f'''(y)$, y el esquema iterativo, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n) f'(y_n)}{f'(x_n) f''(y_n) + f'''(y_n) f(x_n)} \end{cases} \tag{256}$$

De manera más general, si $g(x) = f^{(n)}(y)$, entonces $g'(x) = f^{(n+1)}(y)$, y el esquema iterativo, toma la forma

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n) f^{(n)}(y_n)}{f'(x_n) f^{(n)}(y_n) + f^{(n+1)}(y_n) f(x_n)} \end{cases}$$

Algoritmo 212 *Si la función auxiliar es $g(x_n) = f(x_n) f(y)$ entonces $g'(x_n) = f'(x_n) f(y) + f'(y) f(x_n)$, y el esquema iterativo, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^2(y_n)}{2f'(x_n) f(y_n) + f'(y_n) f(x_n)} \end{cases} \tag{257}$$

Algoritmo 213 *Si la función auxiliar es $g(x_n) = \dfrac{f(x_n)}{f(y)}$ entonces $g'(x_n) = \dfrac{f'(x_n) f(y) - f'(y) f(x_n)}{f^2(y)}$, y el esquema iterativo, toma la forma*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^2(y_n)}{2f'(x_n) f(y_n) - f'(y_n) f(x_n)} \end{cases} \tag{258}$$

Caso 3:

Sabiendo que el método iterativo de Newton está dado por

$$x_{nk+1} = x_n - \frac{h(x_n)}{h'(x_n)}$$

y consideremos $h(x_n) = f(x_n) g(x_n)$ donde $g(x_n) = f(y)$ entonces $h'(x_n) = f'(x_n) f(y) + f'(y) f(x_n)$, el esquema iterativo toma la forma

Algoritmo 214

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n) f(y_n)}{f'(x_n) f(y_n) + f'(y_n) f(x_n)} \end{cases} \tag{259}$$

Caso 4:

Sea la variante del esquema iterativo de Newton

$$x_{n+1} = x_n - \frac{f(x_n) g(h)}{f'(x_n) g(h) + g'(h) f(x_n)}$$

entonces,

Algoritmo 215 *Si la función auxiliar* $g(h) = f'(y)$ *entonces* $g'(h) = f''(y)$ *, y en consecuencia*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = x_n - \dfrac{f(x_n) f'(y_n)}{f'(x_n) f'(y_n) + f''(y_n) f(x_n)} \end{cases} \tag{260}$$

Algoritmo 216 *Sea* $g(h) = f''(y)$ *entonces* $g'(h) = f'''(y)$ *, entonces, el nuevo método iterativo es*

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = x_n - \dfrac{f(x_n) f''(y_n)}{f'(x_n) f''(y_n) + f'''(y_n) f(x_n)} \end{cases} \tag{261}$$

En general, sea $g(h) = f^{(n)}(y)$ entonces $g'(h) = f^{(n+1)}(y)$ en consecuencia

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)} \\ x_{n+1} &= x_n - \frac{f(x_n) f^{(n)}(y_n)}{f'(x_n) f^{(n)}(y_n) + f^{(n+1)}(y_n) f(x_n)} \end{aligned}$$

Caso 5:

De la relación fundamental

$$x_{n+1} = y_n - \frac{f(x_n) f(y_n) g(x_n)}{f'(x_n) [f(x_n) - 2f(y_n)] g(x_n) + f(x_n) f(y_n) g'(x_n)}$$

obtenemos las siguientes variantes del método de Newton.

Algoritmo 217 *Si la función auxiliar* $g(x_n) = f(y)$, *entonces* $g'(x_n) = f'(y)$, *y el esquema iterativo, toma la forma*

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n)[f(y_n)]^2}{f'(x_n)[f(x_n) - 2f(y_n)]f(y_n) + f(x_n)f(y_n)f'(y_n)} \end{array}\right. \tag{262}$$

Algoritmo 218 *Si la función auxiliar* $g(x_n) = f'(y)$, *entonces* $g'(x_n) = f''(y)$, *y el esquema iterativo, toma la forma*

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n)f(y_n)f'(y_n)}{f'(x_n)[f(x_n) - 2f(y_n)]f'(y_n) + f(x_n)f(y_n)f''(y_n)} \end{array}\right. \tag{263}$$

Algoritmo 219 *Si la función auxiliar* $g(x_n) = f''(y)$, *entonces* $g'(x_n) = f'''(y)$, *y el esquema iterativo, toma la forma*

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n)f(y_n)f''(y_n)}{f'(x_n)[f(x_n) - 2f(y_n)]f''(y_n) + f(x_n)f(y_n)f'''(y_n)} \end{array}\right. \tag{264}$$

De forma general, si la función auxiliar $g(x_n) = f^{(n)}(y)$, entonces $g'(x_n) = f^{(n+1)}(y)$, y el esquema iterativo, toma la forma

$$\begin{aligned} y_n &= x_n - \frac{f(x_n)}{f'(x_n)} \\ x_{nk+1} &= y_k - \frac{f(x_n)f(y_n)f^{(n)}(y_n)}{f'(x_n)[f(x_n) - 2f(y_n)]f^{(n)}(y_n) + f(x_n)f(y_n)f^{(n+1)}(y_n)} \end{aligned}$$

Algoritmo 220 *Si la función auxiliar* $g(x_n) = \dfrac{1}{f'(y)}$, *entonces* $g'(x_n) = -\dfrac{f''(y)}{[f'(y)]^2}$, *y el esquema iterativo, toma la forma*

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n)f(y_n)f'(y_n)}{f'(x_n)f'(y_n)[f(x_n) - 2f(y_n)] - f(x_n)f(y_n)f''(y_n)} \end{array}\right. \tag{265}$$

Algoritmo 221 *Si la función auxiliar* $g(x_n) = \dfrac{1}{f(y)}$, *entonces* $g'(x_n) = -\dfrac{f'(y)}{[f(y)]^2}$, *y el esquema iterativo, toma la forma*

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n) f(y_n)}{f'(x_n)[f(x_n) - 2f(y_n)] + f(x_n) f'(y_n)} \end{array}\right. \tag{266}$$

Algoritmo 222 *Si la función auxiliar* $g(x_n) = \dfrac{f(y)}{f'(y)}$, *entonces* $g'(x_n) = -\dfrac{[f'(y)]^2 - f(y) f'(y)}{[f'(y)]^2}$, *y el esquema iterativo, toma la forma*

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n) f'(x_n) f(y_n)}{f'(x_n)[f(x_n) - 2f(y_n)] + f(y_n)\left[[f'(x_n)]^2 - f'(x_n) f''(x_n)\right]} \end{array}\right. \tag{267}$$

Algoritmo 223 *Si la función auxiliar* $g(x_n) = \dfrac{f'(y)}{f(y)}$, *entonces* $g'(x_n) = -\dfrac{f''(y) f(y) - [f'(y)]^2}{[f(y)]^2}$ *, y el esquema iterativo, toma la forma*

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n) f'(x_n) f(y_n)}{f'(x_n)[f(x_n) - 2f(y_n)] + f(y_n)\left[f''(x_n) f'(x_n) - [f'(x_n)]^2\right]} \end{array}\right. \tag{268}$$

Algoritmo 224 *Si la función auxiliar* $g(x_n) = f^2(y)$, *entonces* $g'(x_n) = 2f(y) f'(y)$, *y el esquema iterativo, toma la forma*

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n) f(y_n)}{f'(x_n)[f(x_n) - 2f(y_n)] + 2f(x_n) f'(y_n)} \end{array}\right. \tag{269}$$

Algoritmo 225 *Si la función auxiliar* $g(x_n) = \dfrac{1}{f^2(y)}$, *entonces* $g'(x_n) = -\dfrac{2f'(y)}{f^3(y)}$, *y el esquema iterativo, toma la forma*

$$\left\{\begin{array}{c} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n) f(y_n)}{f'(x_n)[f(x_n) - 2f(y_n)] - 2f(x_n) f'(y_n)} \end{array}\right. \tag{270}$$

9.7. Caso VII : $H(x) = \phi(x) + \lambda (f(\psi(x)) g(\psi(x)))^p$

Ahora vamos a generalizar todos los métodos de técnica variacional, mediante la consideración de funciones auxiliares que envuelven dos funciones predictoras. Las funciones predictoras son $\phi(x_n)$ y $\psi(x_n)$ las cuales tienen orden de convergencia $s \geq 1$ y $t \geq 1$ respectivamente. Estas funciones predictoras permitirán obtener nuevos métodos iterativos de orden $(s+t)$.

Consideremos la forma funcional $H_3(x_n)$ definida por

$$H_3(x) = \phi(x) + \lambda (f(\psi(x)) g(\psi(x)))^p$$

donde

$p = \frac{s}{t}$

$\phi(x_n)$: Función iterativa predictora de orden $s \geq 1$

$\psi(x_n)$: Función iterativa predictora de orden $t \geq 1$

$g(x_n)$: Función arbitraria

λ : Multiplicador de Lagrange

Por otro lado, la condición de optimalidad implica que

$$\lambda = -\frac{1}{[f'(\phi(x_n)) g(\phi(x_n)) + g'(\phi(x_n)) f'(\phi(x_n))]}$$

entonces el esquema iterativo principal se convierte en

$$H(x_n) = \phi(x_n) - \frac{f'(\phi(x_n)) g(\phi(x_n))}{(f'(\phi(x_n)) g(\phi(x_n)) + g'(\phi(x_n)) f(\phi(x_n)))}$$

Es meritorio señalar que el nuevo esquema obtenido incluye al reconocido método de Noor [65] como un caso particular. También notemos que para $\phi(x_n) = x_n$ obtenemos

$$H(x_n) = x_n - \frac{f(x_n) g(x_n)}{f'(x_n) g(x_n) + g'(x_n) f(x_n)}$$

fórmula que fue obtenida en el algoritmo dedicado al método de Newton.

Igual que en los casos anteriores, consideremos la función iterativa definida por el método de Newton

$$\phi\left(x_n\right) = y_n = x_n - \frac{f\left(x_n\right)}{f'\left(x_n\right)}$$

y de esta manera, obtenemos el siguiente esquema iterativo

$$\begin{cases} y_n = x_n - \dfrac{f\left(x_n\right)}{f'\left(x_n\right)} \\ \\ H\left(x_n\right) = y_n - \dfrac{f\left(y_n\right) g\left(y_n\right)}{\left(f'\left(y_n\right) g\left(y_n\right) + g'\left(y_n\right) f\left(y_n\right)\right)} \end{cases} \tag{271}$$

notemos que si dividimos la fracción del esquema anterior por $g\left(y_n\right)$, y si reemplazamos la expresón $\dfrac{g'\left(y_n\right)}{g\left(y_n\right)}$ por $\dfrac{g'\left(x_n\right)}{g\left(x_n\right)}$, obtendremos un esquema iterativo equivalente al anterior, el cual está definido por

$$\begin{cases} y_n = x_n - \dfrac{f\left(x_n\right)}{f'\left(x_n\right)} \\ \\ H\left(x_n\right) = y_n - \dfrac{f\left(y_n\right) g\left(x_n\right)}{\left(f'\left(y_n\right) g\left(x_n\right) + g'\left(x_n\right) f\left(y_n\right)\right)} \end{cases} \tag{272}$$

9.8. Resultados Numéricos

En este acápite se presentarán algunos ejemplos donde se ilustra la eficiencia de los nuevos métodos desarrollados. La comparación se realiza entre el método de Newton y algunos de los algoritmos encontrados. Se tomará como base fundamental el número de iteraciones para encontrar dicha raíz. Se tomaron como conjunto de funciones bases, aquellas funciones que fueron consideradas en diferentes artículos de investigación, las cuales poseen características de ser funciones continuas y diferenciables.

Se han programado todos los algoritmos desarrollados en el lenguaje de alto nivel C++, bajo la filosofía de Programación Orientado a Objeto (POO) y en todos los criterios de parada de los diferentes códigos fuente se utiliza un nivel de tolerancia de $10e^{-16}$.

Se enumeran todos los algoritmos de manera secuencial y se seleccionan aquellos que verifican la condición de ejecutarse en un número menor de iteraciones que el método de Newton y aquellos que logran el mismo número de iteración. Esto se realiza con el objetivo de comparar los algoritmos óptimos y los que son equivalentes (en el número de iteraciones) al método de Newton.

Se aclara que la significación de equivalencia de método en esta tesis, significa que la raíz de la ecuación no lineal converge en un número determinado de iteraciones, aunque su expresión algebraica difiera en la estructura de su fórmula.

Ejemplo 9.1 *Consideremos la ecuacion no lineal* $e^{-x} - x^3 = 0$

Los principales hallazgos respecto a esta ecuación no lineal son :

El método de Newton alcanza la raíz de dicha ecuación en **5** iteraciones, siendo la raíz real con 20 cifras decimales 0,77288295914921012475

Los algoritmos **50, 51, 52, 105, 119, 137 y 177** alcanzan la convergencia de la raíz en **3** iteraciones, superando al método de Newton en dos iteraciones.

Los algoritmos **6, 13, 22, 23, 33, 35, 42, 45, 47, 52, 55, 57, 64, 65, 71, 85, 86, 90, 92, 93, 95, 96, 97, 98, 99, 100, 101, 104, 107, 110, 111, 120, 122, 123, 125, 126, 129, 130, 134, 136, 138, 139, 142, 143, 144, 148, 149, 150, 151, 154, 158, 165, 169, 170, 171 y 178** alcanzan la convergencia de la raíz en 4 iteraciones, superando al método de Newton en una iteración.

Los algoritmos **4, 9, 10, 11, 14, 16, 18, 20, 26, 28, 38, 40, 41, 43, 44, 46, 49, 53, 54, 59, 61, 62, 68, 73, 74, 76, 83, 84, 91, 102, 103, 113, 114, 116, 117, 121, 124, 127, 128, 131, 133, 135, 140, 141, 145, 152, 153, 155, 159, 162, 164, 168, 172 y 175** alcanza la raíz en igual número de iteraciones que el método de Newton, lo cual implica que estos métodos son equivalentes.

El resto de los algoritmos de este capítulo, oscila entre **6 y 13** iteraciones, lo cual significa que no son superiores al método de Newton y que su implementación matemática y computacional no es adecuada y que por tanto, dichos algoritmos no son convenientes para ser utilizado, tanto desde el punto de vista teórico como práctico.

Ejemplo 9.2 *Consideremos la ecuacion no lineal* $x - \cos x = 0$

Los principales hallazgos respecto a esta ecuación no lineal son :

El método de Newton alcanza la raíz de dicha ecuación en **4** iteraciones, siendo la raíz real con 20 cifras decimales 0,73908513321516067229

Los algoritmos **22, 23, 33, 37, 39, 42, 43, 49, 50, 53, 57, 64, 65, 71, 77, 90, 92, 95, 96, 99, 100, 101, 104, 105, 107, 110, 119, 122, 123, 126, 129, 130, 134, 136, 137, 138, 139, 142, 143, 148, 149, 150, 152, 154, 155, 158, 165, 169, 170, 177 y 178** alcanzan la convergencia de la raíz en **3** iteraciones, superando al método de Newton.

Los algoritmos **4, 16, 18, 20, 21, 24, 25, 26, 28, 30, 32, 34, 35, 36, 38, 44, 45, 46, 47, 48, 52, 54, 55, 56, 61, 62, 63, 66, 67, 68, 69, 70, 72, 73, 74, 75, 76, 81, 82, 83, 84, 85, 86, 87, 91, 93, 97, 98, 102, 103, 109, 111, 112, 113, 114, 115, 116, 120, 121, 124, 127, 128, 133, 135, 140, 145, 146, 147, 151, 153, 156, 157, 159, 160, 161, 162, 163, 164, 172, 173, 174 y 175**, alcanza la raíz en igual número de iteraciones que el método de Newton, lo cual implica que estos métodos son equivalentes.

El resto de los algoritmos de este capítulo, oscila entre **5 y 6** iteraciones, lo cual significa que no son superiores al método de Newton y que su implementación matemática y computacional no es adecuada y que por tanto, dichos algoritmos no son convenientes para ser utilizado, tanto desde el punto de vista teórico como práctico.

Ejemplo 9.3 *Consideremos la ecuacion no lineal* $e^{-x} + 2\ln x = 0$

Los principales hallazgos respecto a esta ecuación no lineal son :

El método de Newton alcanza la raíz de dicha ecuación en **5** iteraciones, siendo la raíz real con 20 cifras decimales 0,79851808532225987403

Los algoritmos **51, 96, 100, 105, 119, 122, 136, 137, 139, 142, 144, 149, 169, 171 y 177** alcanzan la convergencia de la raíz en **3** iteraciones, superando al método de Newton.

Los algoritmos **8, 15, 19, 22-24, 26, 33-34, 36-37, 39, 42, 43, 46, 49-50, 52-55, 57, 58, 63-65, 71, 72, 75-77, 82, 84, 90, 92, 94, 95, 97-99, 101, 104, 107, 110, 111, 114, 115, 120, 123, 126, 129-130, 134, 138, 143, 148, 150-155, 157-158, 161, 165, 170, 173 y 178** alcanzan la convergencia de la raíz en 4 iteraciones, superando al método de Newton en una iteración.

Los algoritmos **1, 4, 14, 16, 18, 20, 21, 25, 27, 28, 30, 32, 35, 38, 40, 41, 44, 45, 47, 48, 56, 62, 66, 67, 68, 69, 70, 73, 74, 78, 79, 80, 81, 83, 85, 86, 87, 102, 103, 106, 112, 113, 116, 121, 124, 127, 128, 131, 133, 135, 140, 141, 145, 146, 156, 159, 160, 162, 163, 164, 168, 172 y 175** alcanza la raíz en igual número de iteraciones que el método de Newton, lo cual implica que estos métodos son equivalentes.

El resto de los algoritmos de este capítulo, oscila entre **6 y 11** iteraciones, lo cual significa que no son superiores al método de Newton y que su implementación matemática y computacional no es adecuada y que por tanto, dichos algoritmos no son convenientes para ser utilizado, tanto desde el punto de vista teórico como práctico.

Ejemplo 9.4 *Consideremos la ecuacion no lineal* $x^3 + 4x^2 - 10 = 0$

Los principales hallazgos respecto a esta ecuación no lineal son :

El método de Newton alcanza la raíz de dicha ecuación en **5** iteraciones, siendo la raíz real con 20 cifras decimales 1,36523001341409688791

Los algoritmos **42, 50, 51, 90, 92, 110, 111, 119, 122, 123, 125, 134, 136, 137, 138, 142, 143, 144, 148, 149, 152, 154, 169, 170, 171 y 177** alcanzan la convergencia de la raíz en **3** iteraciones, superando al método de Newton en dos iteraciones.

Los algoritmos **15, 16, 18, 22, 23, 33, 34, 35, 36, 37, 39, 43, 44, 45, 46, 47, 48, 49, 52, 53, 54, 55, 57, 64, 65, 66, 71, 72, 73, 74, 75, 76, 77, 83, 84, 85, 86, 87, 93, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 107, 113, 114, 115, 117, 120, 121, 126, 127, 129, 130, 139, 140, 141, 145, 150, 151, 153, 155, 156, 157, 158, 161, 162, 165, 168, 172 y 174** alcanzan la convergencia de la raíz en **4** iteraciones, superando al método de Newton en una iteración.

Los algoritmos **3, 4, 6, 10, 13, 14, 17, 19, 20, 21, 24, 25, 26, 27, 28, 30, 32, 38, 40, 41, 56, 58, 59, 60, 61, 62, 63, 67, 68, 69, 70, 78, 79, 80, 88, 89, 94, 112, 116, 118, 124, 128, 131, 132, 133, 135, 159, 160, 163, 164, 167, 175 y 176** alcanza la raíz en igual número de iteraciones que el método de Newton, lo cual implica que estos métodos son equivalentes.

El resto de los algoritmos de este capítulo, oscila entre **6 y 10** iteraciones, lo cual significa que no son superiores al método de Newton y que su implementación matemática y computacional no es adecuada y que por tanto, dichos algoritmos no son convenientes para ser utilizado, tanto desde el punto de vista teórico como práctico.

Ejemplo 9.5 *Consideremos la ecuacion no lineal* $e^x - x^3 = 0$

Los principales hallazgos respecto a esta ecuación no lineal son :

El método de Newton alcanza la raíz de dicha ecuación en **4** iteraciones, siendo la raíz real con 20 cifras decimales 1,85718386020783521317

Los algoritmos **22, 23, 33, 34, 36, 37, 39, 42, 43, 46, 49, 50, 51, 52, 53, 54, 55, 57, 64, 65, 71, 72, 73, 75, 76, 77, 84, 85, 90, 92, 93, 95, 96, 97, 98, 99, 101, 104, 105, 107, 110, 111, 114, 115, 119, 122, 123, 126, 129, 130, 134, 136, 137, 138, 139, 142, 143, 144, 148, 149, 150, 151, 152, 153, 154, 155, 157, 158, 161, 162, 165, 169, 170, 171, 177 y 178** alcanzan la convergencia de la raíz en **3** iteraciones, superando al método de Newton en una iteración.

Los algoritmos **1, 2, 3, 4, 6, 8, 10, 13, 14, 15, 16, 17, 18, 19, 20, 21, 24, 25, 26, 27, 28, 29, 30, 32, 35, 38, 40, 41, 44, 45, 47, 48, 56, 58, 59, 60, 61, 62, 63, 66, 67, 68, 69, 70, 74, 78, 79, 80, 82, 83, 86, 87, 88, 89, 91, 94, 102, 103, 108, 112, 113, 116, 117, 118, 120, 121, 124, 125, 127, 128, 131, 132, 133, 135, 140, 141, 145, 156, 159, 160, 163, 164, 166, 167, 168, 172, 175 y 176**, alcanza la raíz en igual número de iteraciones que el método de Newton, lo cual implica que estos métodos son equivalentes.

El resto de los algoritmos de este capítulo, oscila entre **5 y 7** iteraciones, lo cual significa que no son superiores al método de Newton y que su implementación matemática y computacional no es adecuada y que por tanto, dichos algoritmos no son convenientes para ser utilizado, tanto desde el punto de vista teórico como práctico.

9.9. Conclusiones

Las conclusiones referentes al tercer capítulo se pueden sistematizar en las siguientes aseveraciones :

- Aplicación de esquemas iterativos que representan **178** variantes del método de Newton.

- Existencia de nuevos algoritmos desarrollado por la técnica iterativa variacional y que constituyen nuevas variantes del método de Newton, las cuales superan al método clásico de Newton, en una y dos iteraciones.

- Existencia de algoritmos equivalente al método de Newton, referente a la misma cantidad de iteraciones para lograr la raíz real.

- Implementación de **6** casos de la técnica de variación iterativa para la generación de nuevos métodos iterativos.

- Posibilidad de combinar diferentes esquemas iterativos y funciones iterativas para obtener nuevos métodos iterativos que constituyan variantes del método de Newton.

- Todos los algoritmos obtenidos por estas variantes del método de Newton son convergente, oscilando todos ellos entre **3 y 12** iteraciones como máximo.

10. CAPÍTULO V : ANÁLISIS DE CONVERGENCIA

10.1. Convergencia del esquema $H(x) = \phi(x) - \dfrac{\phi'(x)(f(x)g(x))}{p(f'(x)g(x) + g'(x)f(x))}$

En esta sección vamos a analizar el orden de convergencia del esquema iterativo

$$H(x) = \phi(x) - \frac{\phi'(x)(f(x)g(x))}{p(f'(x)g(x) + g'(x)f(x))}$$

probando el siguiente teorema.

Teorema 10.1 *Sea $\phi(x)$ una función de iteración de orden $p \geq 1$, sea r una solución de la ecuación no lineal $f(x)$. Supongamos que $\phi(x)$ es continuamente diferenciable en r, entonces el esquema iterativo anterior tiene orden de convergencia de al menos $p+1$.*

Demostración 11 *Asumamos que*

$$H(x) = \phi(x) - \frac{\phi'(x)(f(x)g(x))}{p(f'(x)g(x) + g'(x)f(x))}$$

además que r es una solución de $f(x)$ y que $\phi(x)$ es una función iterativa convergente de orden p, entonces se tiene que

$$\begin{aligned} f(r) &= 0 \\ \phi(r) &= r \\ \phi'(r) &= 0 \\ &\vdots \\ \phi^{(p-1)}(r) &= 0 \\ \phi^{(p)}(r) &\neq 0 \end{aligned}$$

consideremos a

$$\beta(x) = \frac{f(x)g(x)}{f'(x)g(x) + g'(x)f(x)}$$

entonces

$$\beta(r) = 0$$

porque $f(r) = 0$, usando el hecho de que

$$f(r) = 0 \quad y \quad \beta(r) = 0 \quad y \quad \beta'(r) = 1$$

entonces tenemos que

$$H_1(x) = \phi(x) - \frac{1}{p}\phi'(x)\beta(x)$$

De lo que se deduce que

$$H_1(r) = r$$

ahora realizando derivadas de orden p a la función $H(x)$ obtenemos

$$H^{(p)}(x) = \phi^{(p)}(x) - \frac{1}{p}\sum_{n=0}^{p}\binom{p}{n}\phi^{(p-n+1)}(x)\beta^{(n)}(x)$$

y

$$H^{(p+1)}(x) = \phi^{(p+1)}(x) - \frac{1}{p}\sum_{n=0}^{p+1}\binom{p+1}{n}\phi^{(p-n+2)}(x)\beta^{(n)}(x)$$

de donde se infiere que

$$H^{(p)}(r) = 0$$

porque

$$\phi^{(1)}(r) = \phi^{(2)}(r) = \cdots = \phi^{(p-1)}(r) = 0, \quad \beta(r) = 0\ ,\ \beta'(r) = 0$$

y

$$H^{(p+1)}(r) = -\frac{1}{p}\phi^{(p+1)}(r) \neq 0$$

porque

$$\phi^{(p+1)}(r) \neq 0$$

y esto demuestra que la función iterativa tiene orden de convergencia de al menos $p+1$.

Como un corolario, tenemos que si

$$\frac{f(x)\,g(x)}{f'(x_n)\,g(x) + g'(x_n)\,f(x)} = x - H(x)$$

entonces, la función iterativa $H(x)$

$$H(x) = x - p\frac{x-\phi(x)}{p-\phi'(x)}$$

tiene al menos orden de convergencia $p+1$.

10.2. Convergencia del esquema $x_n = y - \dfrac{f(y)\,g(x)}{f'(x)\,g(x) + g'(x)\,f(x)}$

Teorema 10.2 *Sea la función $f : D \subset R \to R$ que posee una raíz en D, sea $f(x)$ diferenciable en un vecindad de dicha raíz, entonces el algoritmo*

$$x_{n+1} = y_n - \frac{f(y_n)\,g(x_n)}{f'(x_n)\,g(x_n) + g'(x_n)\,f(x)}$$

posee orden de convergencia cúbica y si además la relación se verifica

$$g(x) = -g'(x)\,\frac{f'(x_n)}{f''(x)}$$

entonces el algoritmo posee orden de convergencia cuatro.

Demostración 12 *Consideremos la función asociada con el algoritmo definida por*

$$\psi(x) = y - \frac{f(y)\,g(x)}{f'(x)\,g(x) + g'(x)\,f(x)}$$

Asumiendo que $y(x)$ es cuadráticamente convergente y que r es cero de $f(x)$, entonces

$$\begin{aligned} f(r) &= 0 \\ y(r) &= r \\ y'(r) &= 0 \end{aligned}$$

de aqui

$$\psi(x) = r$$

ahora, derivando

$$\psi'(x) = y' - \frac{f'(y)\,g(x)}{f'(x)\,g(x) + g'(x)\,f(x)} - f(y)\left[\frac{g(x)}{f'(x)\,g(x) + g'(x)\,f(x)}\right]'$$

de donde por las expresiones anteriores

$$\psi'(x) = 0$$

y entonces

$$\begin{aligned} \psi''(x) &= y'' - \frac{f''(y)\,g(x)}{f'(x)\,g(x) + g'(x)\,f(x)} - 2\,[f(y)]'\left[\frac{g(x)}{f'(x)\,g(x) + g'(x)\,f(x)}\right]' \\ &\quad - f(y)\left[\frac{g(x)}{f'(x)\,g(x) + g'(x)\,f(x)}\right]'' \end{aligned}$$

haciendo $x = r$, obtenemos los siguientes resultados

$$\begin{aligned} y''(x) &= \frac{f''(r)}{f'(r)} \\ f''(y) &= f''(r) \\ \frac{g(x)}{f'(x)g(x)+g'(x)f(x)} &= \frac{1}{f'(r)} \\ f(y) &= 0 \\ [f'(y)] &= 0 \end{aligned}$$

y por tanto

$$\psi''(x) = 0$$

Derivando nuevamente $\psi''(x)$ se tiene

$$\begin{aligned} \psi''(x) &= y''' - \frac{f'''(y)g(x)}{f'(x)g(x)+g'(x)f(x)} - [f(y)]''\left[\frac{g(x)f(x)}{f'(x)g(x)+g'(x)f(x)}\right]' \\ &\quad - [f(y)]'\left[\left(\frac{g(x)f(x)}{f'(x)g(x)+g'(x)f(x)}\right)'' + 2\left(\frac{g(x)f(x)}{f'(x)g(x)+g'(x)f(x)}\right)\right] \\ &\quad - [f(y)]\left[2\left(\frac{g(x)f(x)}{f'(x)g(x)+g'(x)f(x)}\right)'' + \left(\frac{g(x)f(x)}{f'(x)g(x)+g'(x)f(x)}\right)'''\right] \end{aligned}$$

nuevamente para $x = r$ se tiene

$$\begin{aligned} \psi'''(x) &= 3\frac{f'''(r)g'(r)}{f'(r)g(r)} + 3\frac{[f''(r)]^2}{[f'(r)]^2} \\ &\neq 0 \end{aligned}$$

por tanto, dicho algoritmo genera al menos un método de orden de convergencia cúbico. Podemos obtener un método de convergencia de orden cuatro si $\psi'''(x) = 0$ y esto es posible si

$$g(r) = -g'(r)\frac{f'(r)}{f''(r)}$$

si sustituimos la expresión anterior en el algoritmo dado, el esquema iterativo toma la forma

$$x_{n+1} = y_n - \frac{f(y_n)f'(x_n)}{[f'(x)]^2 - f(x)f''(x)}$$

y combinando con la relación

$$f''(x_n) = \frac{2f(y)\left[f'(x_n)\right]^2}{\left[f(x_n)\right]^2}$$

obtenemos

$$x_{n+1} = y_n - \frac{f(x_n)}{f'(x_n)}\left[\frac{f(y_n)}{f(x) - 2f(y)}\right]$$

el cual es el método de Ostrowsky el cual tiene orden de convergencia cuatro.

10.3. Convergencia del esquema $x_n = z - \dfrac{f(z)\,g(x)}{f'(x)\,g(x) + g'(x)\,f(x)}$

Teorema 10.3 *Sea la función $f : D \subset R \to R$ que posee una raíz en D, sea $f(x)$ diferenciable en una vecindad de dicha raíz, entonces el algoritmo*

$$\begin{aligned}
y_n &= x_n - \frac{f(x_n)}{f'(x_n)} \\
z_n &= y_n - \frac{f(y_n)}{f'(y_n)} \\
x_{n+1} &= z_n - \frac{f(z_n)\,g(x)}{f'(x_n)\,g(x_n) + g'(x_n)\,f(x_n)}
\end{aligned}$$

posee al menos orden de convergencia quinta

Consideremos la función asociada con el algoritmo definida por

$$\psi(x) = z - \frac{f(z)\,g(x)}{f'(x)\,g(x) + g'(x)\,f(x)}$$

donde

$$\begin{aligned}
y_n &= x_n - \frac{f(x_n)}{f'(x_n)} \\
z_n &= y_n - \frac{f(y_n)}{f'(y_n)}
\end{aligned}$$

Si r es un cero de $f(x)$, y utilizando las relaciones

$$\begin{aligned}
f(r) &= 0 \\
y(r) &= r \\
y'(r) &= 0
\end{aligned}$$

obtenemos que

$$y''(x) = \frac{f''(r)}{f'(r)}$$

y entonces

$$\begin{aligned} z(r) &= r \\ z'(r) &= 0 \\ z''(r) &= 0 \\ z'''(r) &= 0 \end{aligned}$$

y

$$z^{(iv)}(r) = 3\left[\frac{f''(r)}{f(r)}\right]^3$$

de donde, obtenemos

$$\psi(r) = r$$

ahora, derivando

$$\psi'(x) = z' - f'(z)\frac{g(x)}{f'(x)g(x) + g'(x)f(x)} - f(z)\left[\frac{g(x)}{f'(x)g(x) + g'(x)f(x)}\right]'$$

y por las expresiones anteriores

$$\psi'(r) = 0$$

y

$$\psi''(x) = z'' - 2f'(z)\left(\frac{g(x)}{[f(x)g(x)]'}\right)' - f(z)\left[\frac{g(x)}{[f(x)g(x)]'}\right]'' - [f(z)]''\left[\frac{g(x)}{[f(x)g(x)]'}\right]$$

por tanto

$$\psi''(r) = 0$$

ahora, utilizando las relaciones

$$\begin{aligned} f(r) &= 0 \\ y(r) &= r \\ y'(r) &= 0 \end{aligned}$$

y

$$\begin{aligned} z'(r) &= 0 \\ z''(r) &= 0 \end{aligned}$$

se tiene que

$$\psi''(x) = 0$$

Derivando nuevamente $\psi''(x)$ se tiene

$$\begin{aligned}
\psi'''(x) &= z''' - 3f''(z)\left(\frac{g(x)}{f'(x)\,g(x)+g'(x)\,f(x)}\right)' - 3\,[f(z)]'\left[\frac{g(x)}{f'(x)\,g(x)+g'(x)\,f(x)}\right]'' \\
&\quad - f(z)\left(\frac{g(x)\,f(x)}{f'(x)\,g(x)+g'(x)\,f(x)}\right)''' - [f(z)]'''\left(\frac{g(x)}{f'(x)\,g(x)+g'(x)\,f(x)}\right)
\end{aligned}$$

de donde obtenemos que

$$\psi'''(x) = 0$$

nuevamente para $x = r$ se tiene y derivando, obtenemos

$$\begin{aligned}
\psi^{(iv)}(x) &= z^{(iv)} - 4f'''(z)\left(\frac{g(x)}{f'(x)\,g(x)+g'(x)\,f(x)}\right)' - 6\,[f(z)]''\left[\frac{g(x)}{f'(x)\,g(x)+g'(x)\,f(x)}\right]'' \\
&\quad - 4\,[f(z)]'\left(\frac{g(x)\,f(x)}{f'(x)\,g(x)+g'(x)\,f(x)}\right)''' - [f(z)]^{(iv)}\left(\frac{g(x)}{f'(x)\,g(x)+g'(x)\,f(x)}\right) \\
&\quad - f(z)\left(\frac{g(x)}{f'(x)\,g(x)+g'(x)\,f(x)}\right)^{(iv)}
\end{aligned}$$

de donde,

$$\begin{aligned}
\psi^{(iv)}(r) &= 0 \\
z^{(iv)} - f^{(iv)}(z)\left(\frac{g(x)}{f'(x)\,g(x)+g'(x)\,f(x)}\right) &= 0
\end{aligned}$$

ahora simplificando, tenemos

$$\psi^{(v)}(r) = -5\left(\frac{3\,[f''(r)]^3}{[f'(r)]^2}\right)\left(\frac{g'(r)}{f'(r)\,g(r)} - \frac{f''(r)\,g(r)+2f'(r)\,g'(r)}{g(r)\,[f'(r)]^2}\right) \neq 0$$

por tanto, dicho algoritmo genera al menos un método de orden de convergencia quinto.

10.4. Convergencia del esquema $x_n = y - \dfrac{f(x)\,f(y)\,g(x)}{f'(y)\,[f(x)-2f(y)]+f(x)\,f(y)\,g'(x)}$

Ahora probaremos mediante el siguiente teorema, el orden de convergencia del esquema iterativo

$$\left\{\begin{array}{c}
y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\
\\
H(x) = y_n - \dfrac{f(x_n)\,f(y_n)\,g(x_n)}{f'(y_n)\,[f(x_n)-2f(y_n)]+f(x_n)\,f(y_n)\,g'(x_n)}
\end{array}\right. \tag{273}$$

Teorema 10.4 *Sea la función $f : D \subset R \rightarrow R$ que posee una raíz en D, sea $f(x)$ diferenciable en una vecindad de dicha raíz, entonces el algoritmo definido anteriormente posee orden de convergencia cuatro.*

Consideremos a r una raíz simple de f, además $f(x)$ diferenciable en un entorno de r, expandiendo las funciones $f(x)$ y $f'(x)$ en serie de Taylor, alrededor de r, se tiene

$$f(x_n) = f'(r)\left[e_n + c_2e_n^2 + c_3e_n^3 + c_4e_n^4 + c_5e_n^5 + c_6e_n^6 + O\left(e_n^7\right)\right]$$

y

$$f'(x_n) = f'(r)\left[1 + 2c_2e_n + 3c_3e_n^2 + 4c_4e_n^3 + 5c_5e_n^4 + 6c_6e_n^5 + O\left(e_n^7\right)\right]$$

donde

$$c_k = \frac{1}{k!}\frac{f^{(k)}(r)}{f'(r)} \quad , \quad k = 2, 3, \cdots \quad y \quad e_n = x_n - r$$

de las dos expresiones anteriores, obtenemos

$$\begin{aligned} \frac{f(x_n)}{f'(x_n)} &= e_n - c_2e_n^2 + 2\left(c_2^2 - c_3\right)e_n^3 + \left(7c_2c_3 - 4c_2^3 - 3c_4\right)e_n^4 + (8c_2^4 - 20c_3c_3^2 + 6c_3^2 \\ &\quad +10c_2c_4 - 4c_5)e_n^5 + (13c_2c_5 - 28c_2^2c_4 - 5c_6 - 16c_2^5 + 52c_2^3c_3 + 17c_3c_4 \\ &\quad -33c_2c_3^2)e_n^6 + O\left(e_n^7\right) \end{aligned}$$

utilizando la ecuación anterior, tenemos que

$$\begin{aligned} y_n &= r + c_2e_n^2 - 2\left(c_2^2 - c_3\right)e_n^3 - \left(7c_2c_3 - 4c_2^3 - 3c_4\right)e_n^4 - (8c_2^4 - 20c_3c_2^2 + 6c_3^2 \\ &\quad +10c_2c_4 - 4c_5)e_n^5 + (13c_2c_5 - 28c_2^2c_4 - 5c_6 - 16c_2^5 + 52c_2^3c_3 + 17c_3c_4 \\ &\quad -33c_2c_3^2)e_n^6 + O\left(e_n^7\right) \end{aligned}$$

ahora

$$f(y_n) = f'(r)\left[c_2e_n^2 - 2\left(c_2^2 - c_3\right)e_n^3 - \left(7c_2c_3 - 5c_2^3 - 3c_4\right)e_n^4 - (12c_2^4 - 24c_3c_2^2 + 6c_3^2 + 10c_2c_4 - 4c_5)e_n^5 + O\left(e_n^7\right)\right]$$

Efectuando una diferencia funcional entre $f(x) - 2f(y)$ obtenemos

$$f(x) - 2f(y) = f'(r)\left[e_n - c_2e_n^2 + \left(4c_2^2 - 3c_3\right)e_n^3 + (14c_2c_3 - 10c_2^3 - 5c_4)e_n^4 + O\left(e_n^5\right)\right]$$

y

$$f(x)f(y)g(x) = \left[f'(r)\right]^2\left[g(r)c_2e_n^3 + \left\{g'(r)c_2 - g(r)c_2^2 + 2g(r)c_3\right\}e_n^4 + O\left(e_n^5\right)\right]$$

entonces, de las expresiones anteriores, obtenemos

$$
\begin{aligned}
f'(x_n)\left[f(x_n)-2f(y_n)\right]g(x_n)+f(x_n)f(y_n)g'(x_n) \;=\;& \left[f'(r)\right]^2\left[g(r)e_n+\left(g(r)c_2+g'(r)\right)e_n^2\right.\\
&+\frac{1}{2}\left(g''(r)+2g(r)c_2^2\right)e_n^3\\
&+(-\frac{1}{2}g''(r)c_2-2g'(r)c_3+\frac{1}{6}g'''(r)+3g'(r)c_2^2\\
&\left.-2g(r)c_2^3-g(r)c_4+5g(r)c_2c_3)e_n^4+O\left(e_n^5\right)\right]
\end{aligned}
$$

Finalmente

$$
\begin{aligned}
\frac{f(x)f(y)g(x)}{f'(x_n)\left[f(x_n)-2f(y_n)\right]g(x_n)+f(x_n)f(y_n)g'(x_n)} \;=\;& c_2e_n^2+\left(2c_3-2c_2^2\right)e_n^3\\
&+\left(\frac{g'(r)}{g(r)}c_2^2+3c_2^3+c_4-6c_2c_3\right)e_n^4+O\left(e_n^5\right)
\end{aligned}
$$

entonces

$$
x_{n+1}=r+\left(\frac{g'(r)}{g(r)}c_2^2+3c_2^3+c_4-6c_2c_3\right)e_n^4+O\left(e_n^5\right)
$$

donde la ecuación del error está definida por

$$
e_{n+1}=\left(\frac{g'(r)}{g(r)}c_2^2+3c_2^3+c_4-6c_2c_3\right)e_n^4+O\left(e_n^5\right)
$$

y esto completa la demostración de que el esquema iterativo tiene orden de convergencia cuatro.

11. CONCLUSIONES GENERALES

De acuerdo a los objetivos específicos señalados en el presente trabajo, podemos inferir lo siguiente:

Objetivo 1: Generalizar el método de Newton, basado en diferentes enfoque matemáticos y métodos aproximativos para la solución de ecuaciones no lineales.

- Basado en el esquema del método de Newton, se obtuvieron los siguientes resultados por capítulo:

Capítulo 1 : 4 nuevos esquemas y 28 algoritmos.

Capítulo 2 : 19 algoritmos, todos ellos basado en la descomposición de Adomian.

Capítulo 3 : 178 algoritmos, consecuencia de la técnica iterativa variacional.

- En total se construyeron 226 algoritmos variantes del método de Newton.

- Se generalizó el método de Newton mediante la descomposición de los polinomios de Adomian y la técnica iterativa variacional.

- Utilización de la serie de Taylor, la descomposción de Adomian y la técnica iterativa variacional, para la generación de nuevos métodos iterativos.

Objetivo 2: Aplicar la teoría de los polinomios de Adomian y la técnica de variación iterativa para generar nuevos esquemas y métodos iterativos.

- Se generaron nuevos esquemas iterativos basados en la descomposición de los polinomios de Adomian y la técnica iterativa variacional.

- Obtención de nuevos métodos iterativos mediante la aplicación de los métodos de Abbasbandy y Chun.

- Extensión de la fórmula de Abbasbandy mediante los algoritmos AB2 y AB3.

- Generalización del método de Abbasbandy y el método de Chun, mediante la aplicacion de nuevas funciones $N(h)$.

Objetivo 3: Diseñar los algoritmos y programas computacionales de los nuevos métodos iterativos obtenidos, evaluándolo con algunos criterios comparativos y clasificativos.

- Diseño y codificación de todos los algoritmos mediante la aplicacion del lenguaje C++.

- Implementación de la filosofía de Programación Orientada a Objeto en los códigos fuentes de dichos algoritmos.

- Evaluación del código fuente en función del número de iteraciones y evaluaciones funcionales.

- Estudio del índice computacional e índice de eficiencia computacional en la construcción de los algoritmos.

12. REFERENCIAS BIBLIOGRÁFICAS

1. Abbasbandy, S. (2003). Improving Newton-Raphson method for nonlinear equations modified Adomian decomposition method. Appl. Math. Comput. 145, 887–893.

2. Adomian, G. (1989). Nonlinear Stochastic Systems and Applications to Physics. Kluwer Academic Publishers. Dordrecht.

3. Amat, S., Busquier, S. & Gutiérrez, J.M. (2003). Geometric construction of iterative functions to solve nonlinear equations. J. Comput. Appl. Math. 157, 197–205.

4. Babolian, E. & Biazar, J. (2002). Solution of nonlinear equations by modified Adomian decomposition method. Appl. Math. Comput. 132, 162 –172.

5. Basto, M., Semiao, V. & Calheiros, F.L. (2006). A new iterative method to compute nonlinear equations. Appl. Math. Comput. 173, 468–483.

6. Bi, W., Ren, H. & Wu, Q. (2009). Three-step iterative methods with eighth-order convergence for solving nonlinear equations. J. Comput. Appl. Math. 225, 105–112.

7. Bicanic, N. & Johnson, K.H. (1979). Who was "Raphson"? Int. J. Number. Math. Engrg. 14, 148–153.

8. Burden, R.L. & Faires, J.D. (2001). Numerical Analysis. PWS Publishing Company, Bostan, USA.

9. Cajori, F. (1911). Historical note on the Newton-Raphson method of approximation. Amer. Math. Month. 18, 29–32. 148.

10. Chun, C. (2005). Iterative methods improving Newton's method by the decomposition method. Comput. Math. Appl. 50, 1559–1568.

11. Chun, C. (2007). Some improvements of Jarratt's method with sixth-order convergence. Appl. Math. Comput. 190, 1432–1437.

12. Chun, C. (2007). Some variant of Chebshev-Halley method free from second derivative. Appl. Math. Comput. 191, 193–198.

13. Chun, C. (2007). A method for obtaining iterative fórmulas of order three. Appl. Math. Lett. 20, 1103–1109.

14. Chun, C. & Ham, Y. (2003). Some sixth-order variants of Ostrowski root–finding methods. Appl. Math. Comput. 193, 389–394.

15. Chun, C. & Ham, Y. (2008). Some fourth-order modifications of Newton's method. Appl. Math. Comput. 197 (2), 654–658.

16. Chun, C. & Neta, B. (2012). A new sixth-order scheme for nonlinear equations, Appl. Math. Lett. 25, 185–189.

17. Chun, C., Bae, H.J. & Neta, B. (2009). New families of nonlinear third-order solvers for finding multiple roots. Comput. Math. Appl. 57, 1574–1582.

18. Chun, C., Ham, Y. & Lee, S.G (2008). Some higher-order modifications of newton's method for solving nonlinear equations. J. Comput. Appl. Math. 222, 477–486.

19. Dong, C. (1987). A family of multipoint iterative functions for finding multiple roots of equations. Int. J. Comput. Math. 21, 363–367. 149

20. Ezquerro, J.A. & Hernández, M.A. (2003). A uniparametric Halley-type iteration with free second derivative. Int. J. Pure Appl. Math. 6 (1), 103–114.

21. Ezquerro, J.A. & Hernández, M.A. (2004). On Halley-type iterations with free second derivative. J. Comput. Appl. Math. 170, 455–459.

22. Fang, T., Guo, F. & Lee, C.F. (2006). A new iterative method with cubic convergence to solve nonlinear equations. Appl. Math. Comput. 175, 1147–1155.

23. Gary, J., Lastman. & Sinha, N.K. (1989). Microcomputer-Based Numerical Methods for Science and Engineering. New York, NY: W.B. Saunders.

24. Gautschi, W. (1997). Numerical Analysis: An introduction. Birkhauser.

25. Grau, M. & Díaz-Barrero, J.L. (2006). An improvement to Ostrowski root-finding method. Appl. Math. Comput. 173, 450–456.

26. Grau, M. & Díaz-Barrero, J.L. (2006). An improvement of the Euler-Chebyshev iterative method. J. Math. Anal. Appl. 315, 1–7.

27. Grau, M. & Díaz-Barrero, J.L. (2011). Zero-finder methods derived using Runge-Kutta techniques, Appl. Math. Comput. 217, 5366–5376.

28. Grau, M. & Noguera, M. (2004). A variant of Cauchy's method with accelerated fifth order convergence. Appl. Math. Lett. 17, 509–517.

29. Halley, E. (1964). A new, exact and easy method of finding the roots of equations generally and that without any previous reduction, Phil. Roy. Soc. London. 18, 136–145, 150

30. Ham, Y. & Chun, C. (2007). A fifth-order iterative for solving nonlinear equation. Appl. Math. Comput. 194, 287–290.

31. Ham, Y., Chun, C. & Lee, S.G. (2008). Some higher-order modifications of Newton's method for solving nonlinear equations. J. Comput. Appl. Math. 222, 477–486.

32. Hansen, E. & Patrick, M.A. (1977). Family of root finding methods. Numer. Math. 27, 257–269.

33. Hasanov, V.I., Ivanov, I.G. & Nedzhibov, G. (2002). A new modification of Newton method. Appl. Math. Eng. 27, 278–286.

34. He, J.H. (1999). Variational iteration method a kind of nonlinear analytical technique: some examples. Int. J. Nonlinear Mech. 34, 699–708.

35. He, J.H. (2007). Variational iteration method-Some recent results and new interpretations. J. Comput. Appl. Math. 207, 3–17.

36. He, J.H. & Wu, X.H. (2007). Variational iteration method: new development and applications. Comput. Math. Appl. 54 (7–8), 881–894.

37. Homeier, H.H.H. (2003). A modified Newton method for root finding with cubic convergence. J. Comput. Appl. Math. 157, 227–230.

38. Homeier, H.H.H. (2004). A modified Newton method with cubic convergence: the multivariate case. J. Comput. Appl. Math. 169, 161–169.

39. Homeier, H.H.H. (2009). On Newton-type methods for multiple roots with cubic convergence. J. Comput. Appl. Math. 231, 249–254. 151

40. Householder, A.S. (1970). The Numerical Treatment of a Single Nonlinear Equation. McGraw-Hill, New York, USA.

41. Inokuti, M., Sekine H. & Mura, T. (1978). General Use of the Lagrange Multiplier in Nonlinear Mathematical Physics, in: S. Nemat-Nasser (Ed.), Variational methods in the Mechanics of Solids. Pergamon press. New York. pp. 156–162.

42. Jafari, H. & Gejji, V.D. (2006). Revised Adomian decomposition method for solving a system of nonlinear equations. Appl. Math. Comput. 175, 1–7.

43. Javidi, M. (2009). Fourth-order and fifth-order iterative methods for nonlinear algebraic equations. Math. Comput. Model. 50, 66–71.

44. Jovanovic, B. (1972). A method for obtaining iterative fórmulas of higher order. Mat. Vesnik. 9 (24), 365–369.

45. Khan, W. A. (2012). Numerical Techniques for the Solution of Certain Nonlinear Equations. PhD Thesis, COMSATS Institute of Information Technology, Park Road, Islamabad, Pakistán.

46. Khattri, S.K., Noor, M.A. & Al-Said, E. (2011). Unifying fourth-order family of iterative methods. Appl. Math. Lett. 24, 1295–1300.

47. Kim, Y.I. (2010). A new two-step biparametric family of sixth-order iterative methods free from second derivatives for solving nonlinear algebraic equations. Appl. Math. Comput. 215, 3418–3424.

48. King, R. (1973). A family of fourth-order methods for nonlinear equations. SIAM J. Numer. Anal. 10 (5), 876–879. 152

49. Kou, J. (2007). The improvement of modified Newton's method. Appl. Math. Comput. 189, 602–609.

50. Kou, J. & Li, Y. (2007). The improvements of Chebyshev-Halley methods with fifth-order convergence. Appl. Math. Comput. 188(1), 143–147.

51. Kou, J. & Li, Y. (2007). An improvement of the Jarratt method. 189, Appl. Math. Comput. 1816–1821.

52. Kou, J., Li, Y. & Wang, X. (2007). Some new sixth-order methods for solving nonlinear equations. Appl. Math. Comput. 189(1), 647–651.

53. Kou, J., Li, Y. & Wang, X. (2006). Modified Halley's method free from second derivative. Appl. Math. Comput. 183, 704–708.

54. Kou, J., Li, Y. & Wang, X. (2007). A family of fifth-order iterations composed of Newton and third-order methods. Appl. Math. Comput. 186(2), 1258–1262.

55. Kou, J., Li, Y. & Wang, X. (2007). Some variants of Ostrowski's method with seventh-order convergence. J. Comput. Appl. Math. 209, 153–159.

56. Mamta, Kanwar, V., Kukreja, V.K. & Singh, S. (2006). On a class of quadratically convergent iteration formulae. Appl. Math. Comput. 166 (3), 633–637.

57. McNamee, J.M. (2007). Numerical Methods for Roots of Polynomials. Part I, Elsevier, Amsterdam, Holland.

58. Neta, B. (1983). Numerical Methods for the Solution of Equations. Net-A-Sof, California. 153

59. Neta, B. & Johnson, A.N. (2008). High-order nonlinear solver for multiple roots. Comput. Math. Appl. 55, 2012–2017.

60. Noor, K.I. & Noor, M.A. (2007). Predictor-corrector Halley method for nonlinear equations. Appl. Math. Comput. 188, 1587–1591.

61. Noor, K.I. & Noor, M.A. (2007). Iterative methods with fourth-order convergence for nonlinear equations. Appl. Math. Comput. 189, 221–227.

62. Noor, K.I., Noor, M.A. & Momani, S. (2007). Modified Householder iterative methods for nonlinear equations. Appl. Math. Comput. 190, 1534–1539.

63. Noor, M.A. (2007). Fifth-order convergent iterative method for solving nonlinear equations using quadrature fórmula. J. Math. Control Sci. Appl. 1, 241–249.

64. Noor, M.A. (2007). New iterative schemes for nonlinear equations. Appl. Math. Comput. 187, 937–943.

65. Noor, M.A. (2007). New classes of iterative methods for nonlinear equations. Appl. Math. Comput. 191, 128–131.

66. Noor, M.A. (2007). Some iterative methods free from second derivatives for nonlinear equations. Appl. Math. Comput. 192, 101–106.

67. Noor, M.A. (2010). Some iterative methods for solving nonlinear equations using homotopy perturbation method. Int. J. Comput. Math. 87, 141–149.

68. Noor, M.A. (2010). Iterative methods for nonlinear equations using homotopy perturbation technique. Appl. Math. Infomm. Sci. 4, 227–235.

69. Noor, M.A. & Noor, K.I. (2006). Three-step iterative methods for nonlinear equations. Appl. Math. Comput. 183, 322–327.

70. Noor, M.A. & Noor, K.I. (2007). Fifth-order iterative methods for solving nonlinear equations. Appl. Math. Comput. 188, 406–410.

71. Noor, M.A. & Noor, K.I. & Al-Said, E. (2011). New iterative methods for solving integral equations. Int. J. Modern Phys. B. 4655–4660.

72. Noor, M.A. Noor, K.I. & Waseem, M. (2010). Fourth-order iterative methods for solving nonlinear equations. Int. J. Appl. Math. Eng. Sc. 4, 43–52.

73. Noor, M.A. & Shah, F.A. (2009). Variational iteration technique for solving nonlinear equations. J. Appl. Math. Comput. 31, 247–254.

74. Noor, M.A., Shah, F.A., Noor, K.I. & Al-Said, E. (2011). Variational iteration technique for finding multiple roots of nonlinear equations. Sci. Res. Essays. 6(6), 1344–1350.

75. Noor, M. A., Waseem, M. Noor, K. I. & Al-Said, E. (2012).Variational iteration technique for solving a system of nonlinear equations. Optim. Lett. DOI: 10. 1007/s11590–012–0479–3.

76. Osada, N. (1994). An optimal multiple root-finding method of order three. J. Comput. Appl. Math. 51, 131–133.

77. Osada, N. (1998). Improving the order of convergence of iterative functions. J. Comput. Appl. Math. 98, 311–315.

78. Osada, N. (2006). Asymptotic error constants of cubically convergent zero finding methods. J. Comput. Appl. Math. 196, 347–357. 155

79. Osada, N. (2007). A one parameter family of locally quartically convergent zero finding methods. J. Comput. Appl. Math. 205, 116–128.

80. Osada, N. (2008). Chebyshev-Halley methods for analytic functions. J. Comput. Appl. Math. 216, 585–599.

81. Ostrowsky, A.M. (1960). Solutions of Equations and System of Equations. Academic Press, New York.

82. Ostrowsky, A.M. (1973). Solution of Equations in Euclidean and Banach Space. Third ed., Academic Press, New York.

83. Parhi, S.K. & Gupta, D.K. (2008). A sixth order method for nonlinear equations. Appl. Math. Comput. 203, 50–55.

84. Parida, P.K. & Gupta, D.K. (2008). An improved method for finding multiple roots and it's multiplicity of nonlinear equations in R, Appl. Math. Comput. 202, 498–503.

85. Petkovi, M.S. (2010). On a general class of multipoint root-finding methods of high computational efficiency, SIAM. J. Numer. Anal. 47, 4402–4414.

86. Qian, B. (1992). History of Chinese Mathematics. Science Publisher, Beijing.

87. Ramos, J.I. (2008). On the variational iteration method and other iterative techniques for nonlinear differential equations. Appl. Math. Comput. 199, 39–69.

88. Ren, H., Wu, Q. & Bi, W. (2009). New variants of Jarret's method with sixth order convergence. Numer. Alg. 52, 585–603. 156

89. Sargolzaei, P. & Soleymani, F. (2011). Accurate fourteenth-order methods for solving nonlinear equations. Numer. Alg. doi:10.1007/s11075–011–9467–4.

90. Sharma, J.R. & Guha, R.K. (2007). A family of modified Ostrowski methods with accelerated sixth order convergence. Appl. Math. Comput. 190, 111–115.

91. Sharma, J.R., Guha, R.K. & Sharma, R. (2009). Some variants of Hansen–Patrick method with third and fourth order convergence. Appl. Math. Comput. 190, 111–115.

92. Sharma, J.R., Guha, R.K. & Sharma, R. (2011). Improved Ostrowsk like methods based on cubic curve interpolation. Appl. Math. 2, 816–823.

93. Soleymani, F. and Sharifi, M. (2010). A simple but efficient multiple zero-finder for solving nonlinear equations. Far East J. Math. Sci. 42, 153–160.

94. Traub, J.F. (1964). Iterative Methods for the Solution of Equations. Prentice-Hall Englewood Cliffs, New Jersey, USA.

95. Victory, H.D. & Neta, B. (1983). Higher order method for multiple zeros of nonlinear functions. Int. J. Comput. Math. 12, 329–335.

96. Wang, X., Kou, J. & Li, Y. (2006). A varrient of Jarratt method with sixth-order convergence. Appl. Math. Comput. 204, 14–19.

97. Waseem, M. (2012). On Some Iterative Methods for Solving System of Nonlinear Equations. PhD Thesis, COMSATS Institute of Information Technology, Park Road, Islamabad, Pakistán.

98. Weerakoon, S. & Fernando, T.G.I. (2000). A variant of Newton's method with accelerated third-order convergence. Appl. Math. Lett. 13, 87–93. 157

99. Yamamoto, T. (2000). Historical development in convergence analysis for Newton's and Newton-like methods. J. Comput. Appl. Math. 124, 1–23.

100. Zheng, S. & Robbie, D. (1995). A note on the convergence of Halley's method for solving operator equations. J. Austral. Math. Soc. Ser. B. 37, 16–25.

101. Zhou, X., Chen, X. & Song, Y. (2011). Constructing higher-order methods for obtaining the multiple roots of nonlinear equations. J. Comput. Appl. Math. 235, 4199–4206.

13. ANEXO

13.1. Método de Newton

El método clásico para resolver ecuaciones no lineales lo constituye el método de Newton, el cual está definido por los siguientes algoritmos.

Algoritmo 226 *Dado un valor inicial* x_{n0} *se puede aproximar a la solución* x_{n+1} *mediante el esquema iterativo*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} \quad , \quad f'(x_n) \neq 0 \ , \ n = 0, 1, 2, \cdots$$

este método tiene orden de convergencia cuadrática.

El siguiente algoritmo se utiliza para ecuaciones no lineales cuyas raíces poseen un orden de multiplicidad m.

Algoritmo 227 *Dado un valor inicial* x_{n0} *se puede aproximar a la solución* x_{n+1} *mediante el esquema iterativo*

$$x_{n+1} = x_n - m\frac{f(x_n)}{f'(x_n)} \quad , \quad f'(x_n) \neq 0 \ , \ n = 0, 1, 2, \cdots$$

donde m es el orden de multiplicidad de la ecuación no lineal. Este método tiene orden de convergencia cuadrática.

Cuando se desconoce la multiplicidad de los ceros, se puede utilizar el siguiente algoritmo.

Algoritmo 228 *Dado un valor inicial* x_{n0} *se puede aproximar a la solución* x_{n+1} *mediante el esquema iterativo*

$$x_{n+1} = x_n - \frac{f(x_n) f'(x_n)}{[f'(x_n)]^2 - f(x_n) f'(x_n)} \quad , \quad [f'(x_n)]^2 \neq f(x_n) f'(x_n) \ , \ n = 0, 1, 2, \cdots$$

donde m es el orden de multiplicidad del cero de la ecuación no lineal. Para el análisis de convergencia ver $[8, 94]$.

13.2. Polinomios de Adomian

Los polinomios de Adomian están dados por:

$$A_n = \frac{1}{n!}\frac{d^n}{d\lambda^n}\left[N\left(\sum_{i=0}^{\infty}\lambda^i h_i\right)\right]_{\lambda=0} \quad , \quad n = 0, 1, 2, 3, \cdots \tag{274}$$

Los primeros cuatros polinomios de Adomian son

$$\begin{aligned}
A_0 &= N\left(\sum_{i=0}^{\infty}\lambda^i h_i\right)_{\lambda=0} = N(h_0) \\
A_1 &= \frac{d}{d\lambda}\left[N\left(\sum_{i=0}^{\infty}\lambda^i h_i\right)\right]_{\lambda=0} = h_1 N'(h_0) \\
A_2 &= \frac{1}{2}\frac{d^2}{d\lambda^2}\left[N\left(\sum_{i=0}^{\infty}\lambda^i h_i\right)\right]_{\lambda=0} = \frac{1}{2}h_1^2 N''(h_0) + h_2 N'(h_0) \\
A_3 &= \frac{1}{6}\frac{d^3}{d\lambda^3}\left[N\left(\sum_{i=0}^{\infty}\lambda^i h_i\right)\right]_{\lambda=0} = \frac{1}{6}\left[h_1^3 N'''(h_0) + 5h_1h_2N''(h_0) + h_3N'(h_0)\right]
\end{aligned}$$

Los cuales pueden ser expresados en función de las derivadas por

$$\begin{aligned}
h_0 &= \frac{f(x_n)}{f'(x_n)} = c \\
h_1 &= N(h_0) = \frac{h_0^2}{2}\frac{f''(x_n)}{f'(x_n)} = \frac{f^2(x_n)\, f''(x_n)}{2\left[f'(x_n)\right]^3} = A_0 \\
h_2 &= h_1 N'(h_0) = h_1 h_0 \frac{f''(x_n)}{f'(x_n)} = \frac{f^3(x_n)\left[f''(x_n)\right]^2}{2\left[f'(x_n)\right]^5} = A_1 \\
h_3 &= \frac{1}{2}h_1^2 N''(h_0) + h_2 N(h_0) = \frac{3}{8}\frac{f^5(x_n)}{\left[(f'(x_n)\right]^8}\left[f''(x_n)\right]^3 = A_2 \\
h_4 &= \frac{1}{6}\left[h_1^3 N'''(h_0) + 5h_1h_2N''(h_0) + h_3N'(h_0)\right] = \frac{1}{32}\frac{f^5(x_n)}{\left[f'(x_n)\right]^9}\left[f''(x_n)\right]^4 = A_3
\end{aligned}$$

Como $h = c + N(h)$ se tiene

$$\sum_{n=0}^{\infty} h_n = c + \sum_{n=0}^{\infty} A_n$$

Por ser la serie h convergente, se tiene la siguiente importante relación

$$h_0 = c \quad y \quad h_{n+1} = A_n \;\; para\; n \geq 0$$

13.3. Técnica Iterativa Variacional

Los principales casos de la Técnica Iterativa Variacional analizado en el presente trabajo son:

1. $H(x_n) = x_n + \lambda f(x_n)\, g(x_n)$

2. $H(x_n) = \phi(x_n) + \lambda (f(x_n)\, g(x_n))^p$, $p = 1$

3. $H(x_n) = \phi(x_n) - \dfrac{\phi'(x_n)(f(x_n)\, g(x_n))}{(f(x_n)\, g(x_n))'}$

4. $H(x_n) = \phi(x_n) - \dfrac{\phi'(x_n)(f(x_n)\, g(x_n))}{p(f'(x_n)\, g(x_n) + g'(x_n)\, f(x_n))}$

5. $H(x_n) = \phi(x_n) + \lambda (f(x_n)\, g(x_n))^p$, $p = 2$

6. $H(x_n) = \phi(x_n) + \lambda (f(\phi(x_n))\, g(\phi(x_n)))$

7. $H(x_n) = \phi(x_n) + \lambda (f(\psi(x_n))\, g(\psi(x_n)))^p$

13.4. Esquemas Iterativos Complementarios

1. **Algoritmo del Método de Newton-Raphson**

   Dado x_{n0}

   Para $n = 1, 2, \cdots$, hacer $x_{n+1} = x_n - \dfrac{f(x_n)}{f'(x_n)}$, $f'(x_n) \neq 0$

   Generar una sucesión de valores x_{n+1} y verificar la cota de error permisible

2. **Método Iterativo de Abbs**

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n) f''(x_n)}{2[f'(x_n)]^3} - \frac{f^3(x_n)[f''(x_n)]^2}{2[f'(x_n)]^5} \quad ,. \quad n \geq 0$$

3. **Método Iterativo de Chun**

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = x_n - \dfrac{f(x_n)}{f'(x_n)} - 2\dfrac{f(y_n)}{f'(x_n)} + \dfrac{f(y_n) f'(y_n)}{[f'(x_n)]^2} \quad , \quad n \geq 0 \end{cases}$$

4. **Método de Halley**

$$x_{n+1} = x_n - \frac{2f(x_n) f'(x_n)}{2[f'(x_n)]^2 - f(x_n) f''(x_n)}$$

5. **Método de Newton-Halley**

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ z_n = y_n - \dfrac{2f(y_n) f'(y_n)}{2[f'(y_n)]^2 - f(y_n) f''(y_n)} \\ \\ x_{n+1} = z_n - \dfrac{f(z_n)}{f'(z_n)} - \dfrac{[f(z_n)]^2 f''(z_n)}{2[f'(z_n)]^3} \end{cases}$$

6. **Método de HouseHolder (Método 1)**

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)} - \frac{f^2(x_n) f''(x_n)}{2[f'(x_n)]^3}$$

7. **Método de HouseHolder (Método 3)**

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^2(x_n)\, f''(x_n)}{2f'(x)} - \dfrac{f^3(x_n)\left[f''(x_n)\right]^2}{2\left[f'(x_n)\right]^5} \end{cases}$$

8. **Método de Wu**

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(y_n)} \end{cases}$$

9. **Método de Noor y Noor (1 forma)**

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{(y_n - x_n)^2}{2f'(x_n)} f''(x_n) \end{cases}$$

10. **Método de Noor y Noor (3 forma)**

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(y_n)}{f'(y_n)} - \dfrac{f(y_n)\, f''(y_n)}{2\left[f'(y_n)\right]^3} \end{cases}$$

11. **Método Iterativo de Abbasbandy (1 forma)**

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^2(x_n)\, f''(x_n)}{2\left[f'(x_n)\right]^3} - \dfrac{f^3(x_n)\left[f''(x_n)\right]^2}{2\left[f'(x_n)\right]^5} \end{cases}$$

12. **Método Iterativo de Abbasbandy (2 forma)**

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f^2(x_n)\, f''(x_n)}{2f'(x_n)} - \dfrac{f^3(x_n)\, f'''(x_n)}{6\left[f'(x_n)\right]^4} \end{cases}$$

13. **Método de Ostrowski**

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = y_n - \dfrac{f(x_n)}{f'(x_n)} \left(\dfrac{f(y_n)}{f(x_n) - 2f(y_n)} \right) \end{cases}$$

14. **Método de Traub-Ostrowski (Método 1)**

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{n+1} = x_n - \dfrac{f(x_n)}{f'(x_n)} \left(\dfrac{f(y_n) - f(x_n)}{2f(y_n) - f(x_n)} \right) \end{cases}$$

15. **Método de Hueso (Derivada Congelada)**

$$\begin{cases} y_n = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ z_n = y_n - \dfrac{f(y_n)}{f'(y_n)} \\ \\ x_{n+1} = x_n - \dfrac{f(z_n)}{f'(z_n)} \end{cases}$$

16. **Método Iterativo de Chun (1 forma)**

$$\begin{cases} y_k = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{nk+1} = y_k - 2\dfrac{f(y_n)}{f'(x_n)} + \dfrac{f(y_n)\, f'(y_n)}{\left[f'(x_n)\right]^2} \end{cases}$$

17. **Método Iterativo de Chun (2 forma)**

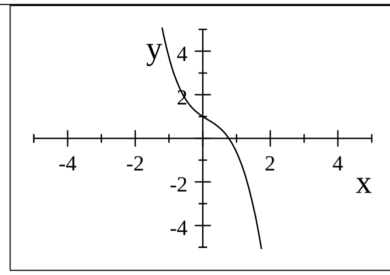
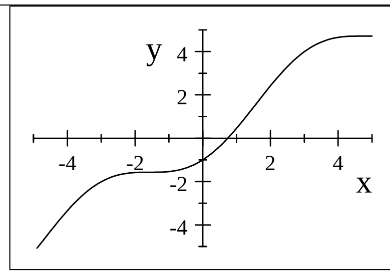
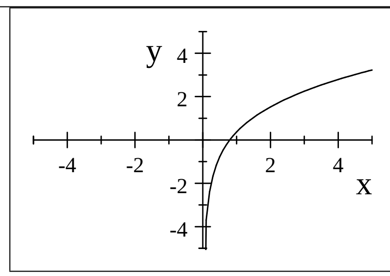
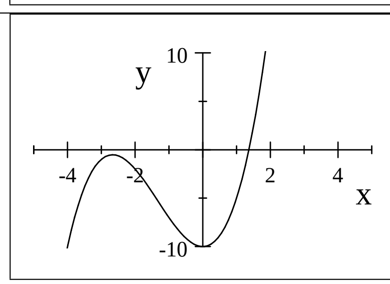
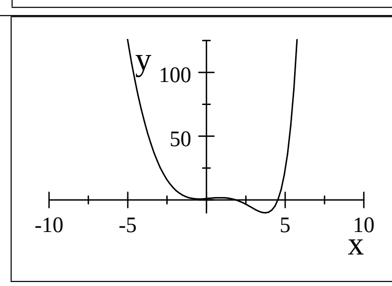
$$\begin{cases} y_k = x_n - \dfrac{f(x_n)}{f'(x_n)} \\ \\ x_{nk+1} = y_k - \dfrac{f(y_n)}{f'(x_n)} \end{cases}$$

13.5. Código Fuente del Algoritmo Newton

```
#include <iostream>
#include <iomanip>
#include <cmath>
#include <cstdlib>
#include <ctime>

using namespace std;

/***************************************************************************/

const double error=10e-09;

class Metodo
{
    double x;
     public:
         double f(double x);
         double d1(double x);
         double NEWTON();
} metodos;

/***************************************************************************/

double Metodo::f(double x)
{
    return(exp(-x)-pow(x,3));
}
/***************************************************************************/

double Metodo::d1(double x)
{
    return(-exp(-x)-3*pow(x,2));
}

/***************************************************************************/
```

```
double Metodo::NEWTON()
{
        long int k=0;
        double x=0.5, Ea, xn;
        cout << endl;
        do
        {
              xn=x-metodos.f(x)/metodos.d1(x);
               Ea=abs((xn-x)/xn);
               x=xn;
              k++;
        } while (Ea>error);
        cout << "Metodo Newton : Iteracion : "<< k << Raiz : "<< fixed << setprecision(20)
<< xn << endl;
        cout << endl;
        return 0;
}

/*************************************************************************/

int main()
{
        metodos.NEWTON();
        metodos.Espacio();
        metodos.Pausa();
        return 0;
}
```

13.6. Gráficas Funcionales

No.	Función	Gráfica
1	$e^{-x_n} - x_n^3$	
2	$x_n - \cos x_n$	
3	$e^{-x_n} + 2\ln x_n$	
4	$x_n^3 + 4x_n^2 - 10$	
5	$e^{x_n} - x_n^3$	

13.7. Raíz obtenida por los algoritmos

No.	Ecuación		Raíz Real
1	$e^{-x_n} - x_n^3$		0,77288295914921012475
2	$x_n - \cos x_n = 0$		0,73908513321516064166
3	$e^{-x_n} + 2\ln x_n = 0$		0,79851808532225992624
4	$x_n^3 + 4x_n^2 - 10 = 0$		1,3652300134140968458
5	$e^{x_n} - x_n^3 = 0$		1,8571838602078353365