

00365



UNIVERSIDAD NACIONAL
AUTONOMA DE MEXICO

Facultad de Ciencias

UN METODO NUMERICO - GRAFICO
PARA LOCALIZAR
CICLOS LIMITE EN EL PLANO

TESIS CON
FALLA DE ORIGEN

T E S I S

Para obtener el grado de:

MAESTRO EN CIENCIAS
(MATEMATICAS)

Presenta:

Miguel López De Luna

1991



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

INDICE

CAPITULO I

TEORÍA BÁSICA DE ECUACIONES DIFERENCIALES ORDINARIAS.

1. Sistemas Autónomos	3
2. Flujo de una Ecuación Diferencial	6
3. Plano Fase	8
4. Clasificación de los Puntos Fijos	10
5. Soluciones Periodicas y Ciclos Límite	15
6. Conjuntos ω - α límite	18

CAPITULO II

SOLUCIÓN DE SISTEMAS DE ECUACIONES NO-LINEALES.

1. Método de Newton para Sistemas de Ecuaciones No Lineales.	23
2. Método de Descenso	27
3. Condiciones de Wolfe-Powell para Convergencia.	31
4. Método de Newton con Búsqueda Lineal	44
5. Método de Broyden	49
6. Análisis de Convergencia del Método de Broyden	55

CAPITULO III

ELECCIÓN DE UN MÉTODO NUMÉRICO PARA RESOLVER ECUACIONES DIFERENCIALES ORDINARIAS.

1. Método de Runge-Kutta	67
2. Formulación General de los Métodos de Runge-Kutta	69
2.1 Control Automático del Paso	78

CAPITULO IV

CÁLCULO DE CICLOS LIMITE

1. Planteamiento del Problema	84
2. Aproximación Por Diferencias	87
3. Métodos de Tiro	87

CAPITULO V

UN SISTEMA PARA CALCULAR CICLOS LIMITE EN EL PLANO

1. Definición	93
2. Bloque Integración y Graficación	93
3. Requerimientos	94
4. Diseño del Sistema	94
4.1 Grafica	94
5. Descripción	100
6. Ejecución del Sistema	118
6.1 El Péndulo	118
6.2 Lista de Comandos de Graficación	123
6.3 Comandos para la Edición de Funciones	124
6.4 Resolviendo $y' = f(x, y)$	125
6.5 Calculando Ciclos Límite	125

CAPITULO VI

EJEMPLOS Y EXPERIENCIA NUMÉRICA

Ejemplo 1.	129
Ejemplo 2.	131
Ejemplo 3.	133
Ejemplo 4.	136
Ejemplo 5.	137
Ejemplo 6.	139
Ejemplo 7.	141

Ejemplo 8.	144
Ejemplo 9.	146
Ejemplo 10.	148
Ejemplo 11.	149

CAPÍTULO I

TEORÍA BÁSICA DE ECUACIONES DIFERENCIALES ORDINARIAS

Sea $t \in \mathbb{R}$, D un subconjunto abierto de un espacio vectorial normado E , una función $f: D \rightarrow E$ y $\dot{x}(t) = \frac{d}{dt}x(t)$. Una ecuación diferencial es de la forma

$$\dot{x}(t) = f(t, x(t)) \quad (1)$$

Decimos que $\phi(t, x)$ es una solución de (1) sobre un intervalo $I \subset \mathbb{R}$ si $\phi(t, x)$ es una función de clase C^1 en I , $(t, \phi(t, x)) \in D$, $t \in I$ y $\phi(t, x)$ satisface (1) para todo $t \in I$.

Geoméricamente, $\phi(t, x)$ es una curva en E cuyo vector tangente $\dot{\phi}(t, x)$ es igual a $f(\phi(t, x))$; imaginamos este vector con origen en $\phi(t, x)$. Nos referiremos a $f: D \rightarrow E$ como un campo vectorial sobre D como se muestra en la figura 1 donde $\phi(t_0, x_0) = x_0$ y $\dot{\phi}(t_0, x_0) = f(x_0)$.

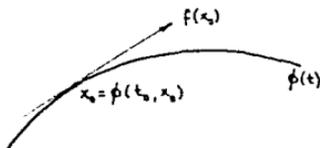


Fig. 1

Supóngase que se tiene $(t_0, x_0) \in D$. Un problema de valor inicial para la ecuación $\dot{x}(t) = f(t, x(t))$ consiste en encontrar un intervalo I conteniendo a t_0 y una solución $\phi(t, x)$ de dicha ecuación que satisface $\phi(t_0, x_0) = x_0$. Se escribe simbólicamente como

$$(CI) \quad \dot{x}(t) = f(t, x(t)), \quad x(t_0) = x_0 \quad t \in I$$

Si existe un intervalo I conteniendo a t_0 y una función $\phi(t, x)$ satisfaciendo (CI) a ésta le llamaremos solución de (1) pasando por el punto (t_0, x_0) .

Una ecuación diferencial puede tener varias soluciones con una condición inicial dada. Por ejemplo, consideremos la ecuación en \mathbb{R}

$$\dot{x}(t) = 3x(t)^{\frac{2}{3}}$$

La función idénticamente cero $\phi(t, x) : \mathbb{R} \rightarrow \mathbb{R}$ dada por $\phi(t, x) = 0$ para todo $t \in \mathbb{R}$ es evidentemente una solución con condición inicial $\phi(0, 0) = 0$. Pero también lo es la función $\phi(t, x) = t^3$. Las gráficas de algunas soluciones se muestran en la figura 2.

Así pues, es claro que para asegurar la unicidad de la solución, es necesario poner condiciones suplementarias sobre la función f . Además de la unicidad de las soluciones, está la cuestión de la existencia; a menudo no es fácil calcular las soluciones explícitamente y tampoco hay garantía de que una ecuación diferencial tenga solución.

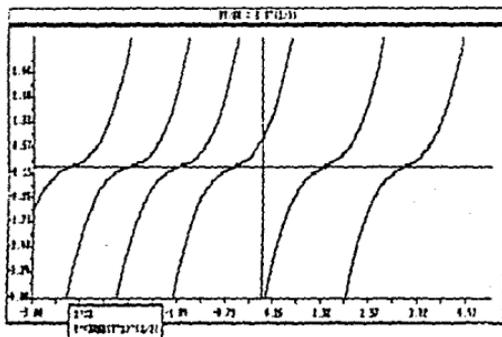


Fig. 2 $x' = x$ $y' = 3xy^{\frac{1}{2}}$

El resultado siguiente es el llamado Teorema Local Fundamental de las ecuaciones diferenciales ordinarias. Se dice que es un teorema "local", porque se refiere a la naturaleza del campo vectorial $f : D \rightarrow E$ en el entorno de un punto $x_0 \in D$. Su demostración, al igual que la de los teoremas presentados hasta la sección II, puede verse en [4].

Teorema

Sean D un subconjunto abierto de un espacio vectorial normado E , $f : W \rightarrow E$ una función C^1 y $x_0 \in W$. Entonces existe un $a > 0$ y una única solución

$$\phi : (-a, a) \rightarrow D$$

de la ecuación diferencial

$$\dot{x}(t) = f(t, x(t))$$

que satisface la condición inicial $\phi(0, x_0) = x_0$

1. Sistemas Autónomos.

La ecuación diferencial

$$\dot{x}(t) = f(t, x(t))$$

es llamada autónoma si la función f no depende de t . En tal caso escribiremos

$$\dot{x} = f(x) \quad (1.1)$$

y de aquí en adelante supondremos que $f: D \rightarrow \mathbb{R}^2$ es de clase C^1 sobre un abierto $D \subseteq \mathbb{R}^2$, así que $\dot{x} = f(x)$ es equivalente a

$$\dot{x}_1 = f_1(x_1, x_2); \quad \dot{x}_2 = f_2(x_1, x_2) \quad (1.2)$$

donde

$$f(x) = \begin{pmatrix} f_1(x_1, x_2) \\ f_2(x_1, x_2) \end{pmatrix} \quad \text{y} \quad x = \begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix}.$$

Una solución para (1.1) consiste de una función

$$\phi(t, x) = \begin{pmatrix} \phi_1(x_1, x_2) \\ \phi_2(x_1, x_2) \end{pmatrix}$$

con $t \in J \subseteq \mathbb{R}$ que satisface (1.2). En general, $\phi(t, x)$ involucra una constante arbitraria. Así que, más que una solución, $\phi(t, x)$ representa una familia de soluciones. En adelante denotaremos $\phi(t, x) = \phi(t)$.

Una consecuencia inmediata del anterior teorema es que dos curvas solución de $\dot{x} = f(x)$ tales que $\phi(t_1) = \psi(t_2)$, entonces $\phi(t_1)$ no es un punto de corte, pues haciendo $\psi_1(t) = \psi(t_2 - t_1 + t)$, ψ_1 es también una solución. Como $\psi_1(t_1) = \psi(t_2) = \phi(t_1)$, resulta que ψ_1 y ϕ coinciden en un entorno de t_1 por la unicidad. Así pues, la situación de la figura 3 no puede darse. Del mismo modo, una curva solución no puede cortarse a sí

misma como en la figura 4.



Fig. 3

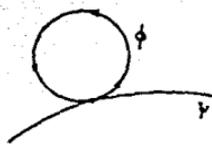


Fig. 4

En el caso en que una curva solución $\phi: J \rightarrow D$ de $\dot{z} = f(x)$ satisface $\phi(t_1) = \phi(t_1 + \omega)$ para ciertos t_1 y $\omega > 0$, entonces dicha curva solución ha de ser cerrada como en la figura 5.



Fig. 5

Para que el anterior teorema sea verdaderamente interesante necesita ser complementado con la propiedad de que la solución $\phi(t)$ depende continuamente de la condición inicial $\phi(t_0)$. Recordemos que una función $f: D \rightarrow E$ con D un subconjunto abierto del espacio vectorial normado E , es Lipschitziana en D , si existe una constante k tal que

$$|f(y) - f(x)| \leq k|y - x| \quad \forall x, y \in D$$

k recibe el nombre de constante de Lipschitz para f .

Teorema

Sea $D \subseteq E$ abierto y supongamos que $f: D \rightarrow E$ tiene una constante de Lipschitz k . Sean $y(t)$ y $z(t)$ dos soluciones de

$$\dot{z} = f(x)$$

en el intervalo cerrado $[t_0, t_1]$. Entonces, para todo $t \in [t_0, t_1]$

$$|y(t) - z(t)| \leq |y(t_0) - z(t_0)|e^{k(t-t_0)}$$

Consideremos nuevamente la ecuación

$$\dot{x} = f(x)$$

con $f: D \rightarrow E$ de clase C^1 , $D \subseteq E$ abierto y supongamos que dos soluciones $u(t)$, $v(t)$ de la anterior ecuación están definidas en el mismo intervalo abierto J que contiene a t_0 y que satisface que $u(t_0) = v(t_0)$, entonces por nuestro anterior teorema de unicidad, $u(t) = v(t)$ en algún intervalo abierto alrededor de t_0 . La unión de todos esos intervalos es el mayor intervalo abierto J^* contenido en J que contiene a t_0 y en el que $u = v$. Pero J^* ha de ser igual a J . Ya que si no sucede así, J^* tendrá un extremo $t_1 \in J$; supongamos que t_1 es el extremo derecho. Por continuidad $u(t_1) = v(t_1)$. Pero nuevamente por nuestro teorema de existencia y unicidad, $u = v$ en cierto intervalo J' que contiene a t_1 . Entonces, $u = v$ en $J^* \cup J' \supseteq J^*$. De donde $u(t) = v(t)$ para toda $t \in J$. Con lo anterior queda demostrado el siguiente resultado:

Lema

Sea $f: D \rightarrow E$ una aplicación C^1 dada. Supongamos que dos soluciones $u(t)$ y $v(t)$ de $\dot{x} = f(x)$ están definidas en el mismo intervalo abierto J que contiene a t_0 y que satisface $u(t_0) = v(t_0)$. Entonces, $u(t) = v(t)$ para todo $t \in J$.

Ahora bien, no está garantizado que una solución $\phi(t)$ de una ecuación diferencial puede definirse para todo t . Por ejemplo, la ecuación en \mathbb{R}

$$\dot{x} = 1 + x^2$$

tiene por soluciones a las funciones

$$\phi(t) = \tan(t - c) \quad c = \text{constante}$$

ya que $\phi(t) \rightarrow \pm\infty$ cuando $t \rightarrow c \pm \frac{\pi}{2}$.

Sea $t_0 \in D$, por el teorema de existencia sabemos que hay un intervalo que contiene a t_0 tal que existe una solución que cumple con $\phi(t_0, x_0) = x_0$. Consideremos a $[\alpha, \beta]$ como la unión de todos los intervalos que contienen a t_0 en los que existe una solución con $\phi(t_0, x_0) = x_0$. Por el lema, las soluciones correspondientes a dos cualesquiera intervalos de la unión coinciden en la intersección de ambos, con lo que existe una solución en todo el intervalo $[\alpha, \beta]$, que recibe el nombre de intervalo maximal donde hay una solución $\phi(t)$ tal que $\phi(t_0) = x_0$.

2. Flujo de una Ecuación Diferencial.

Regresemos a nuestra ecuación $\dot{x} = f(x)$ con $f : D \rightarrow E$ de clase C^1 , $D \subseteq E$ abierto. Sabemos que para cada $x_0 \in E$ existe una única solución $\phi(t, x)$ con $\phi(t_0, x_0) = x_0$ definida en un intervalo abierto maximal $J \subseteq \mathbb{R}$. Denotemos $\phi(t) = \phi(t, x_0)$ para indicar la dependencia de ϕ con respecto a x_0 . Entonces $\phi(t_0, x_0) = x_0$.

Sea $\Omega \subset \mathbb{R} \times W$ el siguiente conjunto:

$$\Omega = \{(t, x) \in \mathbb{R} \times D : t \in J\}$$

La aplicación $(t, x) \mapsto \phi(t, x)$ es entonces una función

$$\phi : \Omega \rightarrow D$$

Se le llama a ϕ flujo de la ecuación $\dot{x} = f(x)$. Lo denotaremos

$$\phi(t, x) = \phi_t(x)$$

Varias preguntas surgen acerca del conjunto Ω y de la función ϕ . ¿Cuáles son las propiedades de Ω ? ¿ ϕ es continua en Ω ? ¿ ϕ tiene derivadas parciales continuas sobre Ω respecto a una o más de sus variables?

Teorema

La aplicación ϕ tiene la siguiente propiedad:

$$\phi_{s+t}(x) = \phi_s(\phi_t(x))$$

en el sentido de que si un miembro de esta igualdad está definido, también lo está el otro y ambos son iguales.

Teorema

Ω es un conjunto abierto en $\mathbb{R} \times D$ y $\phi : \Omega \rightarrow D$ es una función continua.

Consideremos nuevamente la ecuación

$$\dot{x} = 1 + x^2$$

cuyo conjunto solución son las funciones

$$\phi(t) = \tan(t - c) \quad c = \text{constante}$$

y sea $\phi(t_0) = x_0$, entonces

$$\tan(t_0 - c) = x_0$$

de donde

$$c = t_0 - \arctan(x_0)$$

luego la solución a $\dot{x} = 1 + x^2$ que satisface $\phi(t_0) = x_0$ viene dada por

$$\phi(t) = \tan(t - (t_0 - \arctan(x_0)))$$

Observemos que

$$\begin{aligned} \frac{\partial \phi(t)}{\partial x_0} &= \frac{\sec^2(t - (t_0 - \arctan(x_0)))}{1 + x_0^2} \\ &= \frac{1 + \tan^2(t - (t_0 - \arctan(x_0)))}{1 + x_0^2} \\ &= \frac{1 + \phi(t)^2}{1 + x_0^2} \end{aligned}$$

$$\begin{aligned} \frac{d}{dt} \left(\frac{\partial \phi(t)}{\partial x_0} \right) &= \frac{2 \tan(t - (t_0 - \arctan(x_0))) \sec^2(t - (t_0 - \arctan(x_0)))}{1 + x_0^2} \\ &= \frac{2\phi(t)(1 + \phi(t)^2)}{1 + x_0^2} \\ &= 2\phi(t) \left(\frac{\partial \phi(t)}{\partial x_0} \right) \end{aligned}$$

luego $\frac{\partial \phi}{\partial x}(t)$ satisface la ecuación con condición inicial

$$\begin{aligned} \frac{d}{dt} h(t, x) &= f'(\phi(t, x))h(t, x) \\ h(t_0, x) &= 1 \end{aligned}$$

Definición

Sea $\phi(t, x)$ solución de

$$\dot{x}(t) = f(x, x(t)) \quad (2.1)$$

con $f \in C^1(D)$ con D abierto conteniendo la solución $(t, \phi(t, x))$ $0 \leq t < \infty$. El sistema lineal

$$\begin{aligned} \dot{y}(t) &= J_f(t, \phi(t, x))y(t) \\ \phi(t_0, x) &= I \end{aligned}$$

donde J_f e I son las matrices Jacobiana de f e idéntica respectivamente, es llamado Ecuación Varacional asociado a (2.1) respecto a la solución $\phi(t, x)$.

3. Plano Fase.

Analicemos más de cerca la familia de soluciones $\phi(t) = (\phi_1(x), \phi_2(x))^T$ de $\dot{x} = f(x)$ con $f : D \rightarrow \mathbb{R}^2$ de clase C^1 sobre un abierto $D \subseteq \mathbb{R}^2$. El comportamiento cualitativo de esta familia está determinado por el comportamiento de ϕ_1 y ϕ_2 conforme el tiempo t varía, pero en lugar de esto último debemos indicar cómo se comporta ϕ en el plano fase. El plano fase o retrato fase es un subconjunto de \mathbb{R}^2 donde el comportamiento cualitativo de todas las soluciones de $\dot{x} = f(x)$ es representada por una familia de curvas conocidas como trayectorias u órbitas.

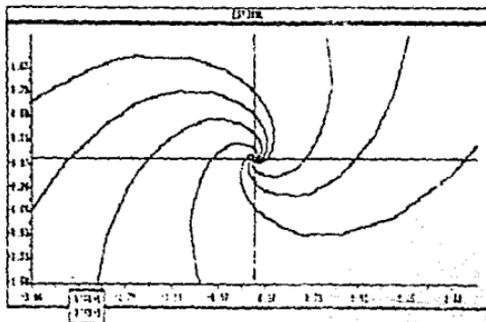


Fig 6 $x' = -y, y' = x + x^2$

Para comenzar a analizar dicha descripción cualitativa en el plano, empezamos fijándonos en los puntos fijos o puntos críticos de (1.2) que son soluciones constantes $\phi(t) = c = (c_1, c_2)$ obtenidas cuando $\phi_1(t) = 0$ y $\phi_2(t) = 0$. Sus correspondientes trayectorias es el punto $\begin{pmatrix} c_1 \\ c_2 \end{pmatrix}$ en el plano fase.

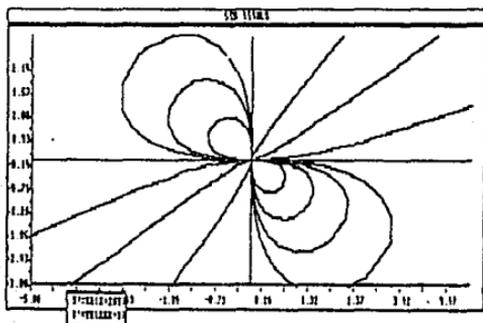


Fig. 7 $x' = x(x+3y)$ $y' = y(2x+y)$

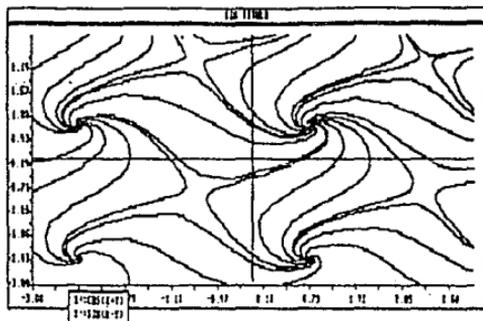


Fig. 8 $x' = \cos(x+y)$ $y' = \text{sen}(x-y)$

Algunos ejemplos son mostrados en las figuras 6,7,8 y 9.

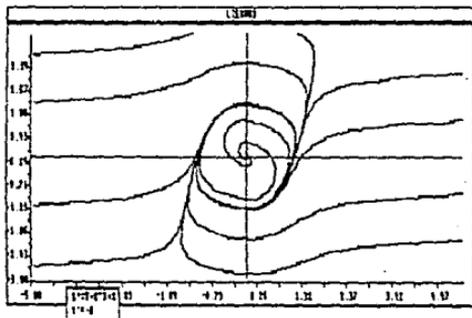


Fig. 9 $x' = y - x^3 + x$ $y' = -x$

4. Clasificación de los Puntos Fijos.

Consideremos nuevamente el sistema (1.1)

$$\dot{x} = f(x) \quad \text{con} \quad x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}, f(x) = \begin{pmatrix} f_1(x) \\ f_2(x) \end{pmatrix}$$

y veamos cómo se comportan las soluciones cerca de sus puntos fijos.

Supongamos que \hat{x} es un punto fijo aislado para el anterior sistema y además que las funciones $f_1(x)$ y $f_2(x)$ son continuas y Lipschitzianas en $D \subseteq \mathbb{R}^2$ tal que \hat{x} pertenezca a D .

- a) Si toda vecindad de \hat{x} contiene una trayectoria cerrada; es decir, una solución $\phi(t)$ de (1.1) tal que $\phi(t_0 + t) = \phi(t_0)$, entonces es llamado un punto de rotación.
- b) Un punto para el cual existe una vecindad que contiene solamente trayectorias

cerradas es llamado centro (figura 10).

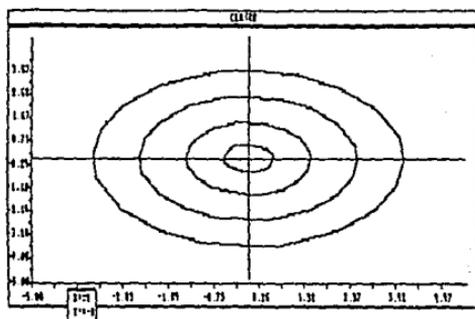


Fig. 10 $x'=y$ $y'=-x$

c) Si existe una vecindad N de \bar{x} con la propiedad de que toda solución que se encuentra en N a partir de cierto t , se aproxima a él conforme $t \rightarrow +\infty$ ($t \rightarrow -\infty$), entonces \bar{x} es llamado un atractor (repulsor). En el caso en que dicha solución sea una espiral, el atractor recibe el nombre de foco (figura 11).

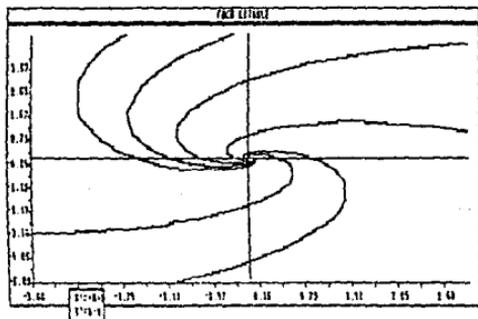


Fig. 11 $x'=-x-y$ $y'=x-y$

d) Si \hat{x} es un atractor y toda línea recta que pasa a través de él es tangente a alguna trayectoria que se aproxima a éste, entonces es llamado un nodo propio (figura 12).

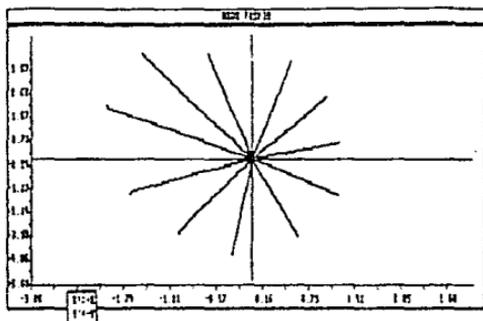


Fig. 12 $x' = -x$ $y' = -y$

e) Si es un atractor y existe una línea recta pasando por \hat{x} , tangente ahí a todas las trayectorias que se le aproximan, entonces dicho punto es llamado un nodo impropio (figura 13).

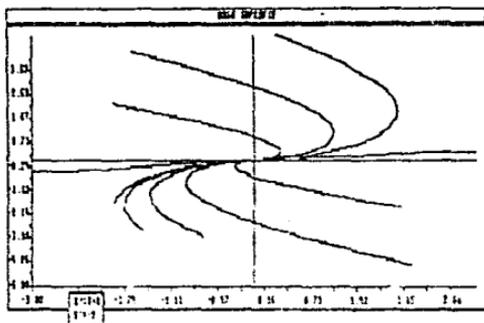


Fig. 13 $x' = y$ $y' = -y$

f) Un punto silla es un punto crítico que es aproximado por sólo un número finito de trayectorias conforme $t \rightarrow \pm\infty$ (figura 14).

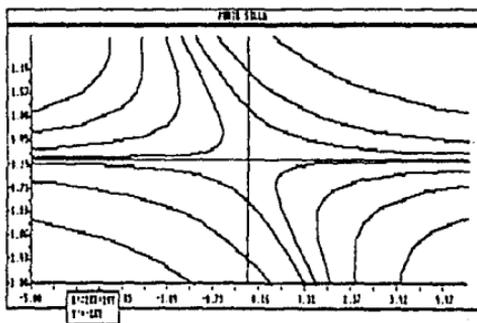


Fig. 14 $x' = 2x + 2y$ $y' = -2y$

g) Todo atractor \hat{x} tiene la propiedad de que toda solución próxima a él permanece próxima en todo tiempo posterior, por lo que también se le llama atractor estable. Estos puntos críticos son de especial interés debido a que el sistema (1.1) puede poseer tales puntos si es lineal.

Teorema

Considere el sistema

$$\begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \end{pmatrix} = \begin{pmatrix} a & b \\ c & d \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$a, b, c, d \in \mathbb{R}$, donde

$$A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

es una matriz no-singular. Sea λ_1, λ_2 los valores propios de A

- 1) Si $\lambda_1 = \alpha + i\beta$ $\lambda_2 = \alpha - i\beta$ con $\beta \neq 0$, entonces el origen es un centro si $\alpha = 0$ y es un foco (foco estable) si $\alpha < 0$ ($\alpha \geq 0$).

- ii) Si la matriz A no tiene dos vectores propios linealmente independientes, entonces el origen es un nodo impropio estable (inestable) si el valor común $\lambda = \lambda_1 = \lambda_2$ es negativo (positivo).
- iii) Si la matriz A es tal que sus valores propios correspondientes a λ_1 y a λ_2 respectivamente, entonces el origen es un nodo propio estable (inestable) si $\lambda_1 = \lambda_2 \leq 0$ ($\lambda_1 = \lambda_2 \geq 0$); el origen es un nodo impropio estable (inestable) si $\lambda_1 \neq \lambda_2$, $\lambda_1 \leq 0$, $\lambda_2 \leq 0$ ($\lambda_1 \geq 0$, $\lambda_2 \geq 0$); y por último el origen es un punto silla si ($\lambda_1 \lambda_2 \leq 0$).

Un punto crítico de un sistema no-lineal puede o no ser de los mencionados anteriormente. Considérese el sistema de la forma siguiente

$$\begin{aligned}\dot{x}_1 &= f_1(x_1, x_2) = ax_1 + bx_2 + P(x_1, x_2) \\ \dot{x}_2 &= f_2(x_1, x_2) = cx_1 + dx_2 + Q(x_1, x_2)\end{aligned}$$

donde $ad - bc \neq 0$.

Si las funciones $P(x_1, x_2)$, $Q(x_1, x_2)$ son suficientemente pequeñas cerca del origen, entonces las trayectorias de este último sistema se asemejarán a las trayectorias del sistema lineal

$$\begin{aligned}\dot{x}_1 &= ax_1 + bx_2 \\ \dot{x}_2 &= cx_1 + dx_2\end{aligned}$$

en una vecindad de dicho punto.

Teorema

Supóngase que el sistema no-lineal

$$\begin{aligned}\dot{x}_1 &= ax_1 + bx_2 + P(x_1, x_2) \\ \dot{x}_2 &= cx_1 + dx_2 + Q(x_1, x_2)\end{aligned}$$

es tal que $ad - bc \neq 0$ y

$$\lim_{|x_1| + |x_2| \rightarrow 0} \frac{|P(x_1, x_2)| + |Q(x_1, x_2)|}{(|x_1| + |x_2|)^{1+t}} = 0$$

para algún $t \geq 0$. Entonces el origen es un nodo, foco o punto silla para ese sistema, si es un nodo, foco o punto silla, respectivamente para el sistema lineal

$$\begin{aligned}\dot{x}_1 &= ax_1 + bx_2 \\ \dot{x}_2 &= cx_1 + dx_2\end{aligned}$$

$$\dot{x}_1 = ax_1 + bx_2 + P(x_1, x_2)$$

$$\dot{x}_2 = cx_1 + dx_2 + Q(x_1, x_2)$$

son igualmente imaginarios puros, entonces el origen puede ser un foco o un centro para el sistema no-lineal. Esto nos lleva a pensar que ciertos sistemas no lineales contienen un número infinito de soluciones periódicas. Este no es siempre el caso como se muestra enseguida.

Consideremos el sistema siguiente

$$\dot{x}_1 = x_2 + x_1 - x_1(x_1^2 + x_2^2)$$

$$\dot{x}_2 = -x_1 + x_2 - x_2(x_1^2 + x_2^2)$$

que es equivalente, usando coordenadas polares, al sistema

$$\dot{r} = r(1 - r^2)$$

$$\dot{\theta} = -1$$

cuya solución es

$$r = \frac{1}{\sqrt{1 + ke^{-2t}}}$$

$$\theta = -t + t_0$$

con k, t_0 constantes

Nótese que el círculo $r = 1$ es una solución; si $k \geq 0$ las correspondientes trayectorias son espirales dirigidas hacia el círculo provenientes de su interior y si $k \leq 0$ las curvas se dirigen hacia el círculo desde su exterior, como se ilustra en la figura 16. En este caso, el círculo es una trayectoria cerrada llamada ciclo límite.

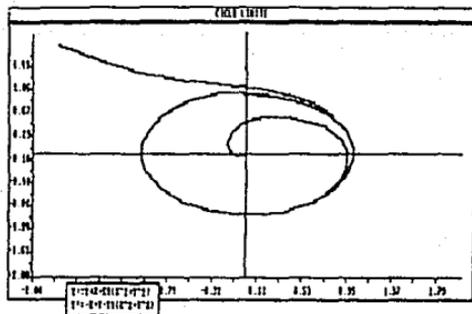


Fig. 16 $x' = y + x - x(x^2 + y^2)$ $y' = -x + y - y(x^2 + y^2)$

Definición.

Una trayectoria cerrada C en un plano fase es llamada un ciclo límite si está aislada de cualquier otra trayectoria cerrada, más precisamente, si existe una vecindad tubular de C que no contiene otras trayectorias cerradas.

No todos los ciclos límite se comportan de la misma forma. Existen tres tipos, a saber:

- i) El ciclo límite estable (atractor) en donde las trayectorias vecinas se aproximan a él en forma de espiral cuando $t \rightarrow +\infty$.
- ii) Se dice que el ciclo límite es inestable (repulsor) cuando las trayectorias vecinas se alejan de él conforme $t \rightarrow +\infty$.
- iii) Por último se tiene el ciclo límite semi-estable, en donde ocurre que sus trayectorias vecinas se aproximan a él por un lado (exterior o interior de dicha solución periódica) y por el otro se alejan como en el ejemplo anterior.

Antes de continuar introduzcamos alguna terminología. Recordemos que habíamos denotado la solución de $\dot{x} = f(x)$ con $x \in \mathbb{R}^2$, que pasa por el punto x_0 en $t = t_0$, por $\phi(t, x_0) = \phi(t)$, ahora la designaremos por el símbolo $\gamma(x_0)$, es decir

$$\gamma(x_0) = \{q \in \mathbb{R}^2 : q = \phi(t, x_0) \ t \in (a, b), -\infty \leq a < 0 < b \leq +\infty\}$$

La semitrayectoria positiva y la negativa de x_0 , se definen reemplazando (a, b) por $(0, b)$ y $(a, 0)$ respectivamente en la igualdad anterior. Las denotaremos por $\gamma^+(x_0)$ y $\gamma^-(x_0)$.

6. Conjuntos ω - α lfmite.

Consideremos el sistema (en coordenadas polares)

$$\begin{aligned} \dot{r} &= r(1-r)(2-r)^2 \\ \dot{\theta} &= 1 \end{aligned}$$

El sistema consta de un único punto fijo, que es el origen; dos trayectorias cerradas C_1 y C_2 ; el resto de ellas son espirales que giran alrededor de las órbitas C_1 y C_2 , como se muestra en la figura 17.

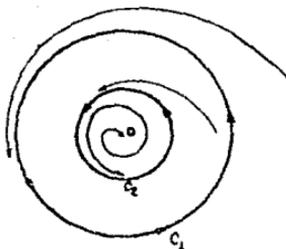


Fig. 17

Ahora bien ¿Cuál es el destino final de un punto p que está en la región exterior a C_1 ? Queremos hacer ver que es toda la curva C_1 .

Sea x_0 fuera de C_1 y q cualquier punto de C_1 . Supongamos que $\phi(t, x_0)$ es la

solución del sistema tal que $\phi(t_0, x_0) = x_0$. La trayectoria correspondiente es la que se muestra en la figura 18. Entonces, en base a lo que se definió anteriormente, decimos que un punto q es destino final de x_0 si la semitrayectoria $\gamma^+(x_0)$ se aproxima a q para una sucesión creciente del tiempo que tiende a $+\infty$. La figura da una idea clara de esto. Es decir, que se puede encontrar una sucesión $\{t_n\} \subset \mathbb{R}^+$ con $t_n \rightarrow +\infty$ y tal que la sucesión $\phi(t_n, x_0) = x_n$ tiene por límite al punto q . Como q era un punto arbitrario de C_1 , esta curva es el conjunto de todos los puntos que son destino final de x_0 .

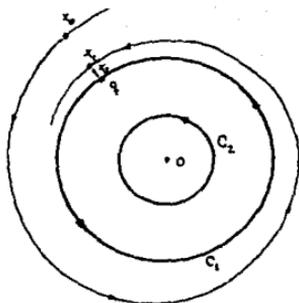


Fig. 18

Haciendo un razonamiento semejante, si x_0 está entre C_1 y C_2 , su destino final cuando t crece sin límite es la curva C_2 (figura 19). En tal caso podemos hacer ver lo siguiente: para cada punto q en C_1 , existe una sucesión $\{t_n\} \subset \mathbb{R}^-$ con $t_n \rightarrow -\infty$, tal que $\lim_{t_n \rightarrow -\infty} \phi(x_0, t_n) = q$. Esto quiere decir que para todo x_0 en la región entre C_1 y C_2 , cada punto sobre C_1 es punto final en el pasado o punto origen de x_0 . Tomando nuevamente x_0 en el exterior de C_1 , vemos que no tiene punto final en el pasado, es decir, cuando $t \rightarrow -\infty$. Finalmente, si tomamos $x_0 \neq O$ en el interior de C_2 , concluimos que su destino final en el futuro es este conjunto; mientras que su destino final en el pasado es el punto O .

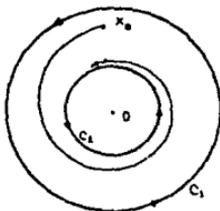


Fig. 19

Resumiendo:

- I) El destino final en el futuro de un punto situado en el exterior de C_2 es esta curva. Este mismo punto no tiene origen en el pasado.
- II) Para un punto x_0 entre C_1 y C_2 , su trayectoria $\gamma(x_0)$ es acotada y tiene tanto destino final en el futuro como en el pasado. Son los conjuntos C_2 y C_1 respectivamente.
- III) La trayectoria γ de un punto en el interior de C_2 es acotada y si el punto es distinto del punto O , su destino final en el futuro es la órbita C_2 y su destino final en el pasado es el punto crítico O .

- iv) Para la trayectoria que pasa por el punto fijo O , su destino final en el futuro y su origen en el pasado es él mismo.

Ahora bien, con lo que se ha dicho en los ejemplos anteriores acerca de los puntos que son el destino final de un punto cuando éste se mueve a lo largo de su trayectoria, se ha hecho evidente la importancia de estos "puntos destino final" en el futuro o en el pasado. Sin embargo, este no es el nombre que reciben. Usualmente a un punto que es el destino final en el futuro se le llama punto límite positivo y a un punto que es el origen final en el pasado se le llama punto límite negativo, también se les conoce como puntos ω -límite y α -límite, respectivamente. Luego la definición formal es la siguiente:

Definición.

Un punto q se llama punto límite positivo o ω -límite de x_0 si existe una sucesión $\{t_n\} \subset \mathbb{R}$ con $t_n \rightarrow +\infty$ tal que $\phi(t_n, x_0) \rightarrow q$. Un punto q se llama límite negativo o α -límite de x_0 si existe una sucesión $\{t_n\} \subset \mathbb{R}$ con $t_n \rightarrow -\infty$ tal que $\phi(t_n, x_0) \rightarrow q$.

Recordemos ahora, que un punto x_0 puede tener a todo un conjunto de puntos como "destino final". De manera natural podemos ahora definir los conjuntos ω -límite y α -límite de un punto x_0 .

Definición.

Sean los conjuntos:

$$A^+(x_0) = \{q : q \text{ es punto } \omega\text{-límite de } x_0\}$$

$$A^-(x_0) = \{q : q \text{ es punto } \alpha\text{-límite de } x_0\}$$

Entonces, $A^+(x_0)$ y $A^-(x_0)$ se llaman, respectivamente conjunto ω -límite (o límite positivo) y conjunto α -límite (o límite negativo) de x_0 .

De la anterior definición se desprende que si x_0 es un punto fijo, entonces es su propio conjunto α -límite y también el ω -límite. Una trayectoria cerrada es el conjunto α -límite y el ω -límite de cada uno de sus puntos.

Existen ejemplos de conjuntos límite que no son ni órbitas cerradas, ni puntos fijos. Por ejemplo, la línea en forma de 8 en el flujo sugerido por la figura 20. Hay tres puntos fijos, dos repulsores y un punto silla. El 8 es el conjunto ω -límite de todos los puntos exteriores a él; la mitad derecha es el conjunto ω -límite de todos los puntos de su interior excepto el punto fijo y análogamente para la mitad izquierda.



Fig. 20

En tres dimensiones existen ejemplos de conjuntos límite sumamente complicados. Sin embargo en el plano son más simples. En los que nosotros estamos interesados, es en los ciclos límite.

Una forma de establecer la existencia de un ciclo límite es resolver directamente el sistema de ecuaciones como en el ejemplo anterior. En general las ecuaciones de un sistema dado son muy difíciles de resolver. Los siguientes resultados nos dan condiciones suficientes para la existencia y no existencia de soluciones periódicas. Al lector interesado en sus demostraciones puede consultar [3].

Teorema (Poincaré-Bendixon)

Si el conjunto ω -límite Ω de una solución para el sistema

$$\begin{aligned}\dot{x}_1 &= f_1(x_1, x_2) \\ \dot{x}_2 &= f_2(x_1, x_2)\end{aligned}$$

es no vacío y no contiene puntos fijos, entonces Ω es una trayectoria cerrada.

Teorema (Criterio Negativo de Bendixon)

Considérese el sistema

$$\begin{aligned}\dot{x}_1 &= f_1(x_1, x_2) \\ \dot{x}_2 &= f_2(x_1, x_2)\end{aligned}$$

donde f_1 y f_2 tienen primeras derivadas parciales continuas en un dominio simplemente conexo D . Si

$$\frac{\partial f_1}{\partial x_1} + \frac{\partial f_2}{\partial x_2}$$

no cambia de signo en D , entonces no existen soluciones periódicas del sistema dentro del conjunto D .

CAPÍTULO II

SOLUCIÓN DE SISTEMAS DE ECUACIONES NO-LINEALES

El problema de calcular los ciclos límite de una ecuación diferencial se va a plantear como un sistema de ecuaciones no-lineales, por lo que nos veremos en la necesidad de resolver éstas numéricamente. El método más usual cuando las derivadas son económicas de evaluar, es el método de Newton y sus modificaciones para que sea globalmente convergente, sin embargo, en nuestro caso las derivadas son muy costosas por lo que necesitaremos métodos que no las usen. Los métodos más efectivos son los de tipo Cuasi-Newton en los que se aproxima el Jacobiano usando unas ideas propuestas por Broyden [2] a fines de 1970. En este capítulo presentamos su método y aspectos teóricos más básicos.

1. Método de Newton para sistemas de ecuaciones no lineales.

Considérese el problema siguiente:

Dada $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$ encontrar x_* en \mathbb{R}^n tal que $F(x_*) = 0$ con $F \in C^1$.

El método de Newton para obtener el punto x_* deseado, consiste en encontrar la raíz a una aproximación a F en la iteración actual x_k . Puesto que

$$F(x_k + p) = F(x_k) + \int_{x_k}^{x_k+p} J(x) dx$$

donde J = Jacobiano de F

aproximando la integral $\int_{x_k}^{x_k+p} J(x) dx$ por $J(x_k)p$ se obtiene la aproximación a F en x_k con perturbación p

$$F(x_k + p) \approx M_k(x_k + p) = F(x_k) + J(x_k)p$$

y resolviendo para s_k tal que $M_k(x_k + s_k) = 0$ se obtiene

$$\begin{aligned} J(x_k)s_k &= -F(x_k) \\ x_{k+1} &= x_k + s_k \end{aligned}$$

Ejemplo

Sea

$$F(x) = \begin{pmatrix} x_1 + x_2 - 3 \\ x_1^2 + x_2^2 - 9 \end{pmatrix}$$

que tiene raíces $\begin{pmatrix} 3 \\ 0 \end{pmatrix}$ y $\begin{pmatrix} 0 \\ 3 \end{pmatrix}$, y sea $x_0 = \begin{pmatrix} 1 \\ 5 \end{pmatrix}$. Entonces tenemos

$$J(x) = \begin{pmatrix} 1 & 1 \\ 2x_1 & 2x_2 \end{pmatrix}$$

$$J(x_0)s_0 = -F(x_0);$$

$$\begin{pmatrix} 1 & 1 \\ 2 & 10 \end{pmatrix} s_0 = -\begin{pmatrix} 3 \\ 17 \end{pmatrix};$$

$$s_0 = \begin{pmatrix} -\frac{13}{8} \\ -\frac{11}{8} \end{pmatrix}$$

$$x_1 = x_0 + s_0$$

$$= \begin{pmatrix} 1 \\ 5 \end{pmatrix} + \begin{pmatrix} -\frac{13}{8} \\ -\frac{11}{8} \end{pmatrix}$$

$$= \begin{pmatrix} -0.625 \\ 3.625 \end{pmatrix}$$

$$J(x_1)s_1 = F(x_1);$$

$$\begin{pmatrix} 1 & 1 \\ -\frac{5}{4} & \frac{29}{4} \end{pmatrix} s_1 = \begin{pmatrix} 0 \\ \frac{145}{32} \end{pmatrix};$$

$$s_1 = \begin{pmatrix} \frac{145}{272} \\ -\frac{145}{272} \end{pmatrix}$$

$$x_2 = x_1 + s_1$$

$$= \begin{pmatrix} -0.625 \\ 3.625 \end{pmatrix} + \begin{pmatrix} \frac{145}{272} \\ -\frac{145}{272} \end{pmatrix}$$

$$\approx \begin{pmatrix} -0.092 \\ 3.092 \end{pmatrix}$$

El método de Newton trabaja bien en el ejemplo anterior; x_2 ya está cerca de la solución $\begin{pmatrix} 0 \\ 3 \end{pmatrix}$.

Esta es la ventaja principal del método: si x_0 está lo suficientemente cerca de la solución x_* y $J(x_*)$ es no-singular, se probará posteriormente que la sucesión $\{x_k\}$ generada por el algoritmo anterior converge cuadráticamente a x_* . En el caso en que x_0 no esté cercano a x_* , el método no necesariamente converge a x_* , como se muestra en la figura 1, lo cual muestra que el x_0 elegido no es un buen comienzo.

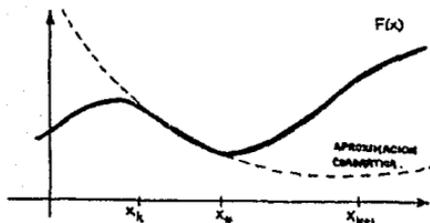


Fig. 1

Existen otros dos problemas fundamentales en la implementación computacional del algoritmo anterior. Primero, no se puede disponer en la mayoría de los casos, de una expresión analítica del Jacobiano $J(x)$ de $F(x)$, por ejemplo cuando $F(x)$ misma no se conoce analíticamente. Así que tenemos que aproximar $J(x)$ por diferencias finitas o por algún otro método. Segundo, $J(x_k)$ puede ser una matriz singular o mal-condicionada, lo cual dificulta calcular s_k a partir de $J(x_k)s_k = -F(x_k)$.

Para resolver los problemas anteriores con el método de Newton, empezaremos por hacer ver que la convergencia local del método de Newton se puede globalizar usando algunas variantes de los métodos de descenso en optimización no-lineal.

El siguiente teorema establece una de las principales ventajas del método de Newton, a saber, la convergencia cuadrática, cuya demostración puede verse en [2].

Teorema (Convergencia Local del Método de Newton)

Sea $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$ de clase C^1 en un conjunto abierto convexo $D \subset \mathbb{R}^n$. Supongamos que existen $x_* \in \mathbb{R}^n$, $r, \beta > 0$ tal que $B(x_*, r) \subset D$, $F(x_*) = 0$, además, que exista la inversa del Jacobiano evaluada en x_* , $J^{-1}(x_*)$ tal que $\|J^{-1}(x_*)\| \leq \beta$ y $J \in Lip_\gamma(B(x_*, r))$. Entonces, existe $\epsilon > 0$ tal que para todo $x_0 \in B(x_*, \epsilon)$ la sucesión x_1, x_2, \dots dada por $x_{k+1} = x_k - J^{-1}(x_k)F(x_k)$ $k = 0, 1, 2, \dots$ está bien definida, converge a x_* y

$$\|x_{k+1} - x_*\| \leq \beta\gamma \|x_k - x_*\|^2 \quad k = 0, 1, 2, \dots$$

Algoritmo de Newton

Dada $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$ de clase C^1 y $x_0 \in \mathbb{R}^n$.

Para $k = 0, 1, 2, \dots$

Resolver

$$J(x_k)s_k = -F(x_k)$$

$$x_{k+1} = x_k + s_k$$

La idea básica para construir un algoritmo más eficaz para ecuaciones no-lineales, es combinar una estrategia globalmente convergente con otra más rápida que sea localmente convergente, de tal manera que haga uso de ambas bondades. Para ello usaremos el Algoritmo de Newton en cada iteración, si otorga un paso razonable s_k , lo usaremos y de no ser así, tomamos s_k dado por una estrategia global. Dicho método siempre nos llevará a usar Newton cerca de la solución buscada y así recuperar la convergencia cuadrática. Si el método global es escogido e incorporado adecuadamente, el algoritmo también será globalmente convergente. Todo algoritmo que hace uso de la idea anterior recibe el nombre de Cuasi-Newton.

A continuación presentaremos las ideas básicas de los métodos de descenso que nos permitirán diseñar estrategias globalmente convergentes y empezaremos por explicar cómo podemos transformar el problema que tenemos en un problema de optimización.

Supóngase que la aproximación x_k no se encuentra "suficientemente cerca" de la solución x_* , luego ¿bajo qué criterio podríamos decidir si se acepta al punto x_{k+1} como la nueva aproximación? Una idea razonable, es aceptarlo si $\|F(x_{k+1})\|_2 < \|F(x_k)\|_2$, donde $\|F(x)\|_2^2 = F^T(x)F(x)$. Así, resolver $F(x) = 0$ nos sugiere obtener el $\min_{x \in \mathbb{R}^n} f(x)$ con

$$f: \mathbb{R}^n \rightarrow \mathbb{R} \quad f(x) = \frac{1}{2} \|F(x)\|_2^2$$

donde " $\frac{1}{2}$ " se usa por conveniencia algebraica.

Nótese que una solución a $F(x) = 0$ es un mínimo global de $f(x)$, pero f puede tener un mínimo global x_* tal que $F(x_*)$ no tenga solución, además de que como en los métodos para encontrar mínimos, por lo general se localizan mínimos locales y un mínimo local de f no necesariamente es solución de

$F(x) = 0$ como lo muestra la siguiente figura

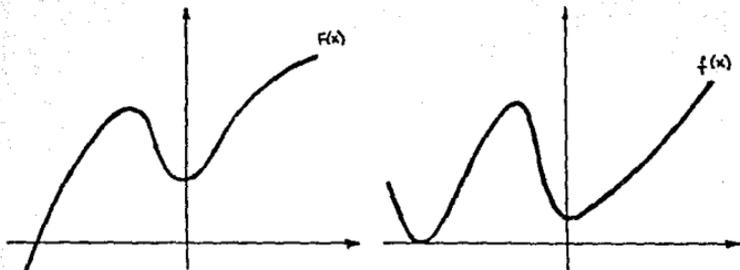


Fig. 2

Para evitar lo anterior, se usará una estrategia global para encontrar la solución a $F(x) = 0$, que haga uso de la estructura del problema original lo más posible y que estará basada en un análisis también global para solucionar $\min_{x \in \mathbb{R}^n} f(x)$. Por lo anterior, de ahora en adelante, para resolver el problema $F(x) = 0$ consideraremos el problema de optimización asociado, dado por

$$\min \frac{1}{2} \|F(x)\|_2^2.$$

A continuación veremos los métodos globales para el cálculo de mínimos locales de funciones de varias variables y al final veremos cómo se aplican para construir una versión globalmente convergente del método de Newton.

2. Métodos de descenso

La forma más usual de saber en qué dirección decrece $f: \mathbb{R}^n \rightarrow \mathbb{R}$ si se está en un punto x_0 , es mediante el uso del gradiente, que como sabemos, nos indica la dirección de más rápido crecimiento de la función a partir del punto en cuestión. Por consiguiente, la dirección opuesta al gradiente nos da la mayor velocidad de descenso para una cierta vecindad de x_0 . Ésto se puede ver de la siguiente manera: consideremos cómo varía la función f a lo largo de dirección p_k , es decir, consideremos la función

$$\phi_k(\alpha) = f(x_0 + \alpha p_k)$$

donde x_0 y p_k son fijos. Lo que queremos entonces es hacer ver que la pendiente de la curva $\phi_k(\alpha)$ en $\alpha = 0$ es más negativa si escogemos p_k en la dirección opuesta al

gradiente (figura 3). Para ello, observemos que, por la regla de la cadena, tenemos

$$\phi'_k(\alpha) = \nabla^T f(x_0 + \alpha p_k) p_k$$

de donde

$$\phi'_k(0) = \nabla^T f(x_0) p_k = \|\nabla f(x_0)\| \|p_k\| \cos \theta_k$$

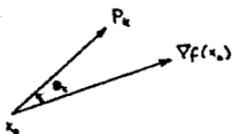


Fig. 3

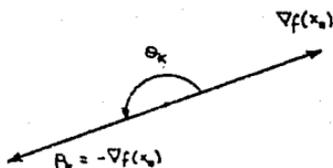


Fig. 4

Si nos fijamos ahora en todas las p_k posibles con la misma norma del gradiente, resulta que el valor mínimo de $\phi'_k(0)$ se obtiene cuando

$$p_k = -\nabla f(x_0)$$

ya que en este caso, $\cos \theta_k = -1$ como lo muestra la figura 4.

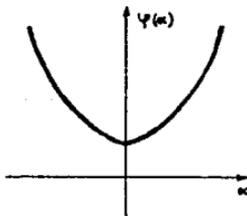


Fig. 5

Nótese que si tomamos una dirección \bar{p} ortogonal a $\nabla f(x_0)$, entonces la gráfica de $\phi(\alpha) = f(x_0 + \alpha \bar{p})$ sería como lo muestra la figura 5. Lo anterior nos muestra pues, que el gradiente en un punto dado es de gran ayuda en tanto que nos indica las direcciones en las que es factible esperar que la función f disminuya.

Así tenemos que, si deseamos que se cumpla

$$f(x_0 + \alpha p) < f(x_0)$$

para valores pequeños positivos de α , es necesario pedir que

$$p^T \nabla f(x_0) < 0$$

esto es, debe pedirse que el vector p forme un ángulo mayor de 90° con el gradiente $\nabla f(x_0)$.

Definición

Un vector p se dice que es una dirección de descenso en x_0 si forma un ángulo mayor de 90° con $\nabla f(x_0)$, es decir, si $p^T \nabla f(x_0) < 0$.

Cabe hacer las siguientes observaciones:

- 1) Aparentemente, de las posibles direcciones a lo largo de las cuales es de esperarse un decremento de la función f a partir de x_0 , la mejor de ellas es $-\nabla f(x_0)$, ya que ésta nos indica justamente la dirección donde la función disminuye más rápidamente. Sin embargo, resulta que lo anterior no es necesariamente cierto, puesto que esta propiedad sólo es válida para una cierta vecindad de x_0 . Esto se puede ilustrar geométricamente como sigue

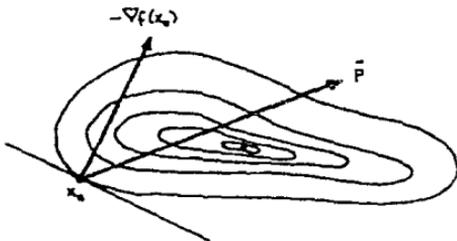


Fig. 6

En este caso, como se observa, \bar{p} nos lleva mucho más cerca del mínimo que $-\nabla f(x_0)$. Sin embargo, si decidimos tomar la dirección $-\nabla f(x_0)$ y para evitar el alejarnos demasiado del mínimo buscado, debemos tomar como dirección un múltiplo

adecuado de $-\nabla f(x_0)$. En ocasiones se elige $-\lambda \nabla f(x_0)$; $\lambda > 0$ de tal forma que $\|-\lambda \nabla f(x_0)\| = 1$, es decir, $\lambda = \frac{1}{\|\nabla f(x_0)\|}$.

- II) La función $\phi(\alpha) = f(x_0 + \alpha p)$ a lo largo de una dirección de descenso siempre está acotada y tiene por lo menos un mínimo local si f está acotada.
- III) Los mínimos y máximos locales de $\phi(\alpha)$ corresponden a los puntos donde la dirección p de descenso es tangente a una curva de nivel de f .

Hasta ahora el único paso que hemos considerado para el problema de minimización, es la dirección dada por el método de Newton

$$s_k = -\nabla^2 f^{-1}(x_k) \nabla f(x_k).$$

Por lo tanto, es natural preguntarse si la dirección de Newton es una dirección descendente, lo que será cierto si y sólo si

$$\nabla^T f(x_k) s_k = -\nabla^T f(x_k) \nabla^2 f^{-1}(x_k) \nabla f(x_k) < 0$$

que es verdad si $\nabla^2 f^{-1}(x_k)$ o, equivalentemente $\nabla^2 f(x_k)$ es positiva definida. Esto garantiza que para pasos suficientemente pequeños en la dirección de Newton, el valor de $f(x)$ decrecerá.

Algoritmo "Método del descenso más rápido".

Dada $f: \mathbb{R}^n \rightarrow \mathbb{R}$ de clase C^1 , $x_0 \in \mathbb{R}^n$

Para $k = 1, 2, \dots$

Encontrar α_k que resuelva

$$\min_{\alpha_k > 0} f(x_k - \alpha_k \nabla f(x_k))$$

$$x_{k+1} = x_k - \alpha_k \nabla f(x_k)$$

Este no es un método numérico recomendable, debido a que cada iteración involucra una minimización unidimensional exacta y que su convergencia es lineal y, en ocasiones muy lentamente, pues la constante de convergencia lineal depende de la condición de $\nabla^2 f$ en el óptimo y entre mayor es la iteración más lento el método. Afortunadamente lo anterior no sucede para todas las direcciones de descenso y no es necesario realizar minimizaciones exactas para tener convergencia como mostramos a continuación.

3. Condiciones de Wolfe-Powell para Convergencia.

Dada una dirección de descenso p_k , debemos tomar un adecuado α_k que nos lleve a obtener un aceptable x_{k+1} . Es decir

en la iteración k :

hacer $x_{k+1} = x_k + \alpha_k p_k$ para un adecuado $\alpha_k > 0$ tal que x_{k+1} se encuentre más cerca de la solución que x_k .

Gráficamente, esto quiere decir que x_{k+1} será escogida en la dirección en la cual f decrezca a partir de x_k como se observa en la siguiente figura

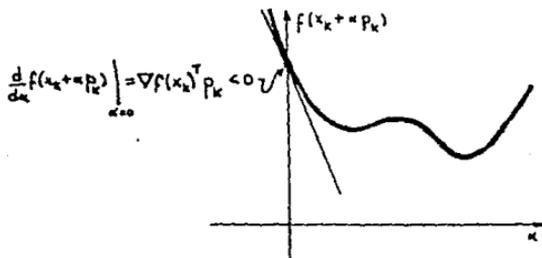


Fig. 7

donde p_k es cualquier dirección descendente.

El procedimiento más común es comenzar con $\alpha_k = 1$ y en caso de que este paso no nos lleve a un x_{k+1} aceptable, hacemos una reducción adecuada de α_k para poder

obtener un posible x_{k+1} , como se ilustra enseguida

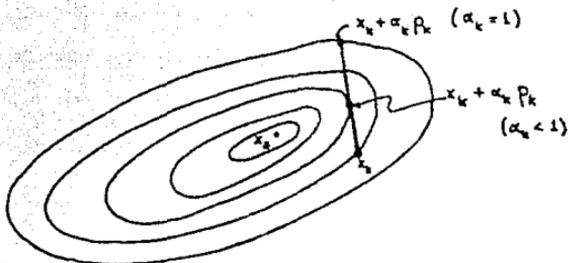


Fig. 8

en que $f(x_k + \alpha_k p_k) < f(x_k + p_k)$ con $\alpha_k < 1$ lo que nos lleva a perder la convergencia cuadrática del método de Newton cerca de la solución que corresponde al valor $\alpha_k = 1$. Por consiguiente, es importante que el procedimiento para escoger α_k conduzca a elegir $\alpha_k = 1$ cerca del mínimo buscado.

El sentido común nos hace pensar que x_{k+1} se encuentra más cerca que x_k de algún mínimo local de f cuando

$$f(x_{k+1}) < f(x_k)$$

Pero para sorpresa nuestra, la anterior condición no garantiza que la sucesión $\{x_k\}$ convergerá a un mínimo de f como se muestra en el siguiente ejemplo.

Ejemplo

Sea $f(x) = x^3$ $x_0 = 2$

$$\{p_k\} = \{(-1)^{k+1}\} \quad \{\alpha_k\} = \{2 + 3(2^{-(k+1)})\}$$

luego

$$x_1 = x_0 + \alpha_0 p_0 = -\frac{3}{2}; \quad x_2 = x_1 + \alpha_1 p_1 = \frac{5}{4}; \dots$$

$$\{x_k\} = \{(-1)^k (1 + 2^{-k})\}$$

cada p_k es una dirección descendente en x_k ya que

$$\begin{aligned} f'(x_k)p_k &= (2x_k)(-1)^{k+1} \\ &= 2(-1)^k(1+2^{-k})(-1)^{k+1} \\ &= -2(1+2^{-k}) < 0 \end{aligned}$$

además $f(x_{k+1}) < f(x_k)$ con

$$\lim_{k \rightarrow \infty} f(x_k) = \lim_{k \rightarrow \infty} (1+2^{-k})^2 = 1.$$

Para ilustrar lo anterior se da la siguiente figura.

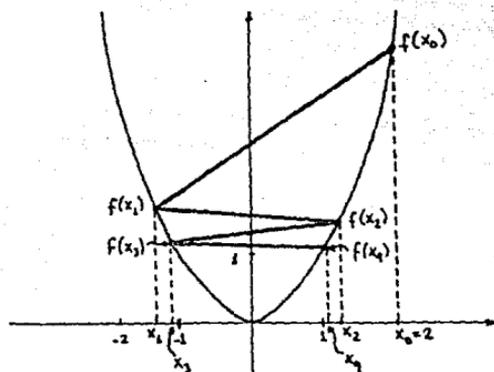


Fig. 9

Por suerte $f(x_k)$ no tiene como límite a un mínimo de f y además $\{x_k\}$ tiene como puntos límite a ± 1 , así que no converge.

Ahora consideremos la misma función y condición inicial $x_0 = 2$ y tomemos $\{p_k\} = \{-1\}$, $\{\alpha_k\} = \{2^{-k+1}\}$, entonces

$$\{x_k\} = \left\{2, \frac{3}{2}, \frac{5}{4}, \frac{9}{8}, \dots\right\} = \{1+2^{-k}\}$$

nuevamente

$$\begin{aligned} f'(x_k)p_k &= (2x_k)(-1) \\ &= -2(1+2^{-k}) < 0 \end{aligned}$$

así que p_k es una dirección descendente y también

$$f(x_{k+1}) = 1 + 2^{-(k+1)} < 1 + 2^{-k} = f(x_k)$$

en este caso

$$\lim_{k \rightarrow \infty} x_k = 1$$

que no es un mínimo de f como lo muestra la siguiente figura

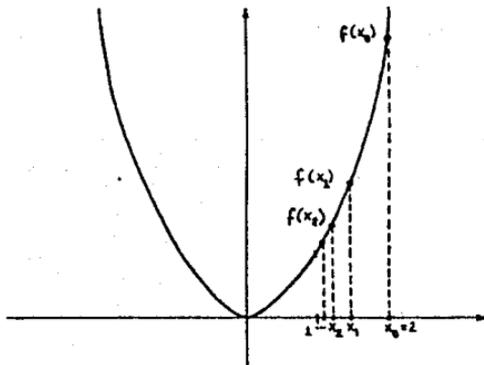


Fig. 10

El problema en el primer caso es que se obtenían pequeños decrecimientos de f relativos a los valores correspondientes de α . Podemos remediar esto requiriendo que el decrecimiento desde $f(x_k)$ a $f(x_{k+1})$ sea al menos alguna fracción dada del decrecimiento inicial en esa dirección; es decir, sea $\lambda \in (0, 1)$ y escójase α de todas las $\alpha > 0$ que satisfagan

$$f(x_{k+1}) \leq f(x_k) + \alpha \lambda \nabla^T f(x_k) p \quad (1)$$

cada p_k es una dirección descendente en x_k ya que

$$\begin{aligned} f'(x_k)p_k &= (2x_k)(-1)^{k+1} \\ &= 2(-1)^k(1+2^{-k})(-1)^{k+1} \\ &= -2(1+2^{-k}) < 0 \end{aligned}$$

además $f(x_{k+1}) < f(x_k)$ con

$$\lim_{k \rightarrow \infty} f(x_k) = \lim_{k \rightarrow \infty} (1+2^{-k})^2 = 1$$

Para ilustrar lo anterior se da la siguiente figura.

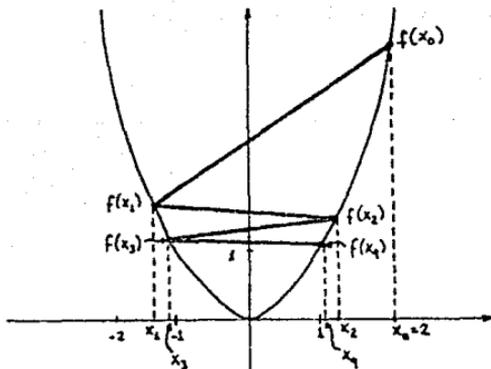


Fig. 9

Por suerte $f(x_k)$ no tiene como límite a un mínimo de f y además $\{x_k\}$ tiene como puntos límite a ± 1 , así que no converge.

Ahora consideremos la misma función y condición inicial $x_0 = 2$ y tomemos $\{p_k\} = \{-1\}$, $\{\alpha_k\} = \{2^{-k+1}\}$, entonces

$$\{x_k\} = \left\{2, \frac{3}{2}, \frac{5}{4}, \frac{9}{8}, \dots\right\} = \{1+2^{-k}\}$$

nuevamente

$$\begin{aligned} f'(x_k)p_k &= (2x_k)(-1) \\ &= -2(1+2^{-k}) < 0 \end{aligned}$$

como se observa a continuación

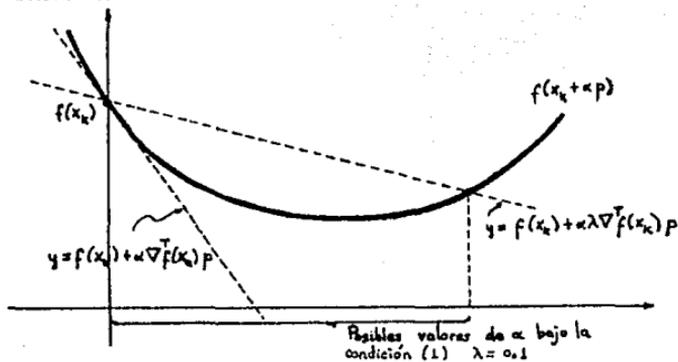


Fig. 11

Lo anterior es equivalente a

$$f(x_{k+1}) \leq f(x_k) + \lambda \nabla^T f(x_k)(x_{k+1} - x_k) \quad (1')$$

ya que $x_{k+1} = x_k + \hat{\alpha}p$.

El problema en el segundo caso del ejemplo, es que los pasos son muy pequeños relativos al decrecimiento inicial de f . Para evitar lo anterior, debemos de requerir que el decrecimiento de f en la dirección p en el punto x_{k+1} sea mayor que alguna fracción del decrecimiento en la dirección p en el punto x_k , es decir

$$\nabla^T f(x_{k+1})p = \nabla^T f(x_k + \hat{\alpha}p)p \geq \beta \nabla^T f(x_k)p \quad (2)$$

o equivalentemente

$$\nabla^T f(x_{k+1})(x_{k+1} - x_k) \geq \beta \nabla^T f(x_k)(x_{k+1} - x_k) \quad (2')$$

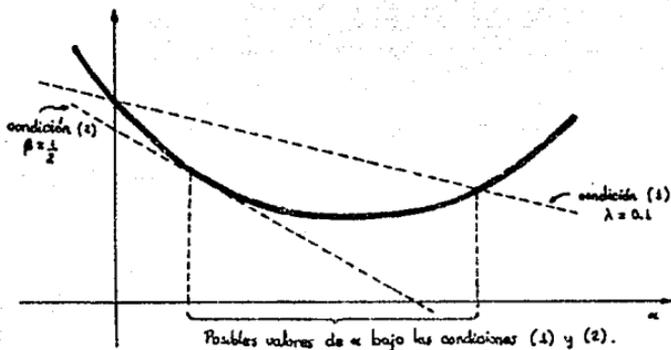


Fig. 12

La condición $\beta > \lambda$ garantiza que las condiciones (1) y (2) se satisfacen simultáneamente (figura 12).

El siguiente teorema muestra que para cualquier dirección descendente p_k , existen puntos $x_k + \alpha_k p_k$ satisfaciendo las condiciones (1) y (2), que reciben el nombre de condiciones de Wolfe-Powell para la Convergencia.

Teorema

Sea $f: \mathbb{R}^n \rightarrow \mathbb{R}$ de clase C^1 sobre \mathbb{R}^n . Sea $x_k \in \mathbb{R}^n$, $p_k \in \mathbb{R}^n$ tal que $\nabla^T f(x_k)p_k < 0$ y supóngase que el conjunto

$$\{f(x_k + \alpha p_k) : \alpha > 0\}$$

está acotado inferiormente. Entonces, si $0 < \lambda < \beta < 1$, existen $\alpha_u > \alpha_l > 0$ tal que $x_{k+1} = x_k + \alpha_k p_k$ satisface las condiciones de Wolfe-Powell si $\alpha_k \in (\alpha_l, \alpha_u)$.

Demostración:

Como $\lambda < 1$ se tiene que

$$f(x_k + \alpha p_k) < f(x_k) + \lambda \alpha \nabla^T f(x_k) p_k$$

para toda $\alpha > 0$ suficientemente pequeña, ya que $f(x_k + \lambda p_k)$ también está acotada inferiormente, entonces existe $\hat{\alpha} > 0$ más pequeño tal que

$$f(\hat{x}) = f(x_k + \hat{\alpha} p_k) = f(x_k) + \hat{\alpha} \lambda \nabla^T f(x_k) p_k \quad (*)$$

Así que cualquier $x \in (x_k, \hat{x})$ cumple con la condición (1). Por el teorema del valor medio, existe $\bar{\alpha} \in (0, \hat{\alpha})$ tal que

$$\begin{aligned} f(\hat{x}) - f(x_k) &= \nabla^T f(x_k + \bar{\alpha} p_k) (\hat{x} - x_k) \\ &= \nabla^T f(x_k + \bar{\alpha} p_k) \bar{\alpha} p_k \end{aligned} \quad (**)$$

De (*) y (**)

$$\nabla^T f(x_k + \bar{\alpha} p_k) p_k = \lambda \nabla^T f(x_k + \hat{\alpha} p_k) p_k > \beta \nabla^T f(x_k) p_k$$

puesto que $\lambda < \beta$ y $\nabla^T f(x_k) p_k < 0$. Por la continuidad de ∇f , la desigualdad anterior sigue siendo válida para α en algún intervalo (α_l, α_u) conteniendo a $\bar{\alpha}$. Por consiguiente, si $(\alpha_l, \alpha_u) \subset (0, \hat{\alpha})$, entonces $x_{k+1} = x_k + \alpha_k p_k$ satisface las condiciones de Wolfe-Powell para $\alpha_k \in (\alpha_l, \alpha_u)$

□

Teorema

Sea $f: \mathbb{R}^n \rightarrow \mathbb{R}$ acotada de clase C^1 en \mathbb{R}^n y supóngase que existe $\gamma \geq 0$ tal que

$$\|\nabla f(z) - \nabla f(x)\|_2 \leq \gamma \|z - x\|_2$$

para todo $x, z \in \mathbb{R}^n$. Entonces, dado cualquier $x_0 \in \mathbb{R}^n$ existe una sucesión $\{x_k\}$ $k = 0, 1, 2, \dots$ que satisface las condiciones de Wolfe-Powell para la cual si $\nabla f(x_k) \neq 0$ para toda k , entonces se tiene

$$(3) \quad \lim_{k \rightarrow \infty} \frac{\nabla^T f(x_k) s_k}{\|s_k\|_2^2} = 0$$

donde $s_k = x_{k+1} - x_k$

Demostración:

Como $\nabla f(x_k) \neq 0$ entonces existe p_k tal que $\nabla^T f(x_k)p_k < 0$ (por ejemplo $p_k = -\nabla f(x_k)$). Por el teorema anterior, existe $\alpha_k > 0$ tal que $x_{k+1} = x_k + \alpha_k p_k$ satisface las condiciones de Wolfe-Powell. Supongamos sin pérdida de generalidad que $\|p_k\|_2 = 1$, luego $\|s_k\|_2 = \|x_{k+1} - x_k\|_2 = \alpha_k \|p_k\|_2 = \alpha_k$. Debemos probar que si $s_k \neq 0$ para toda k , entonces $\sigma_k \rightarrow 0$ donde

$$\sigma_k = \frac{\nabla^T f(x_k)s_k}{\|s_k\|_2^2}$$

por (1') y del hecho $\alpha_k \sigma_k < 0$ para toda k , tenemos para $j > 0$

$$\begin{aligned} f(x_j) - f(x_0) &= \sum_{k=0}^{j-1} f(x_{k+1}) - f(x_k) \\ &\leq \sum_{k=0}^{j-1} \lambda \nabla^T f(x_k)s_k \\ &= \lambda \sum_{k=0}^{j-1} \alpha_k \sigma_k \\ &< 0 \end{aligned}$$

luego, como f está acotada, se tiene

$$\sum_{k=0}^{\infty} \alpha_k \sigma_k < \infty$$

entonces

$$\lim_{k \rightarrow \infty} \alpha_k \sigma_k = 0.$$

Para concluir que

$$\lim_{k \rightarrow \infty} \sigma_k = 0$$

usamos la condición (2')

$$\nabla^T f(x_{k+1})s_k \geq \beta \nabla^T f(x_k)s_k$$

y así

$$[\nabla f(x_{k+1}) - \nabla f(x_k)]^T s_k \geq (\beta - 1) \nabla^T f(x_k)s_k > 0$$

luego $\beta < 1$, por Cauchy-Schwartz y

$$\|\nabla f(z) - \nabla f(x)\|_2 \leq \gamma \|z - x\|_2$$

tenemos

$$\begin{aligned} \gamma \|x_{k+1} - x_k\|^2 &\geq \|\nabla f(x_{k+1}) - \nabla f(x_k)\|_2 \|s_k\| \\ &\geq (\nabla f(x_{k+1}) - \nabla f(x_k))^T s_k \geq (\beta - 1) \nabla^T f(x_k) s_k \end{aligned}$$

$$\gamma \alpha_k^2 = \gamma s_k^2 \geq (\beta - 1) \nabla^T f(x_k) s_k = (\beta - 1) \alpha_k \sigma_k$$

así que

$$\begin{aligned} \alpha_k &\geq \frac{\beta - 1}{\gamma} \sigma_k > 0 \\ \alpha_k \sigma_k &\leq \frac{\beta - 1}{\gamma} \sigma_k^2 < 0 \end{aligned}$$

así

$$0 = \lim_{k \rightarrow \infty} \alpha_k \sigma_k \leq \frac{\beta - 1}{\gamma} \lim_{k \rightarrow \infty} \sigma_k^2 \leq 0$$

lo que muestra que

$$\lim_{k \rightarrow \infty} \sigma_k = 0$$

□

Nótese que mientras el teorema anterior es válido para cualquier algoritmo de búsqueda sobre la línea, es independiente del método para seleccionar α_k . Además, aunque el teorema también garantiza que

$$\frac{\nabla f^T(x_k) s_k}{\|s_k\|_2} = \|\nabla f(x_k)\| \cos \theta_k \rightarrow 0$$

no necesariamente se cumple $\nabla^T f(x_k) \rightarrow 0$ a menos que $\theta_k < \frac{\pi}{2} - \epsilon$ para algún $\epsilon > 0$ y para toda k , lo que se debe de tomar en cuenta al elegir la dirección de descenso y así Wolfe-Powell nos garantiza que $\|\nabla f(x_k)\| \rightarrow 0$.

Ahora resolvamos el problema de calcular un α_k que satisfaga las condiciones impuestas. Comenzaremos con $\alpha_k = 1$ y si $x_k + \alpha_k p_k$ no es buena aproximación reduciremos α_k hasta encontrar un aceptable $x_k + \alpha_k p_k$. Debido a que la condición (2) de

Wolfe-Powell en ocasiones nos lleva a obtener α_k demasiado pequeña, no haremos uso de ella.

Algoritmo "Estrategia para calcular α_k "

Dada $\lambda \in (0, \frac{1}{2})$ $0 < l < u < 1$

en la iteración k tomar

$$\alpha_k = 1$$

Mientras

$$f(x_k + \alpha_k p_k) > f(x_k) + \lambda \alpha_k \nabla^T f(x_k) p_k \text{ hacer}$$

$$\text{i) } \alpha_k = \rho \alpha_k \text{ para algún } \rho \in [l, u]$$

$$\text{ii) } x_{k+1} = x_k + \alpha_k p_k$$

En la práctica λ es muy pequeña, generalmente se usa $\lambda = 10^{-4}$.

La estrategia anterior será útil para construir un algoritmo que calcule el tamaño del paso α_k . A continuación desarrollaremos los detalles necesarios para obtener un método eficiente.

Sea $\hat{f}(\alpha) = f(x_k + \alpha p_k)$, si $\alpha = 1$ no satisface la condición (1), necesitamos disminuirla. Para ello usaremos la información previa de \hat{f} :

$$\hat{f}(0) = f(x_k), \quad \hat{f}'(0) = \nabla^T f(x_k) p \quad \text{y} \quad \hat{f}(1) = f(x_k + p_k)$$

como $\hat{f}(1) = f(x_k + p_k)$ no satisface la condición (1), es decir, si

$$\hat{f}(1) = f(x_k + p_k) > f(x_k) + \lambda \nabla^T f(x_k) p = \hat{f}(0) = \lambda \hat{f}'(0)$$

aproximamos $\hat{f}(\alpha)$ por una cuadrática usando $\hat{f}(0)$, $\hat{f}'(0)$ y $\hat{f}(1)$:

$$m_{OUA}(\alpha) = (\hat{f}(1) - \hat{f}(0) - \hat{f}'(0)) \alpha^2 + \hat{f}'(0) \alpha + \hat{f}(0)$$

calculamos el punto $\hat{\alpha}$ para el cuál $m'_{OUA}(\hat{\alpha}) = 0$

$$m'_{OUA}(\hat{\alpha}) = 2(\hat{f}(1) - \hat{f}(0) - \hat{f}'(0))\hat{\alpha} + \hat{f}'(0) = 0$$

$$\hat{\alpha} = -\frac{\hat{f}'(0)}{2(\hat{f}(1) - \hat{f}(0) - \hat{f}'(0))}$$

$$m''_{OUA}(\hat{\alpha}) = 2(\hat{f}(1) - \hat{f}(0) - \hat{f}'(0)) > 0$$

ya que

$$\hat{f}(1) > \hat{f}(0) + \lambda \hat{f}'(0) > \hat{f}(0) + \hat{f}'(0)$$

Así que $\hat{\alpha}$ minimiza $m_{OUA}(\alpha)$ (figura 13), también $\hat{\alpha} > 0$ ya que, $\hat{f}'(0) < 0$. Por lo tanto, haremos $\alpha_x = \hat{\alpha}$ en el algoritmo

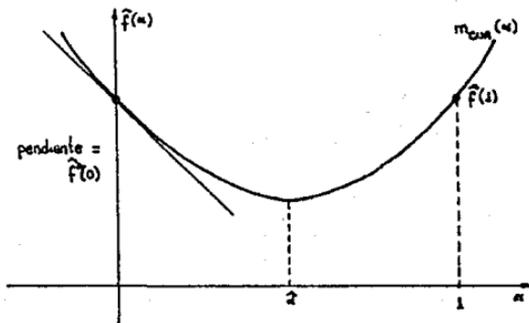


Fig. 13

Nótese que como $\hat{f}(1) > \hat{f}(0) + \lambda \hat{f}'(0)$, tenemos

$$\begin{aligned} \hat{\alpha} &= -\frac{\hat{f}'(0)}{2(\hat{f}(1) - \hat{f}(0) - \hat{f}'(0))} \\ &< \frac{-\hat{f}'(0)}{2(\lambda \hat{f}'(0) - \hat{f}'(0))} \\ &= -\frac{1}{2(\lambda - 1)} \end{aligned}$$

Si $\hat{f}(1) \geq \hat{f}(0)$, entonces

$$\hat{\alpha} = -\frac{\hat{f}'(0)}{2(\hat{f}(1) - \hat{f}(0) - \hat{f}'(0))} \leq -\frac{\hat{f}'(0)}{2(-\hat{f}'(0))} = \frac{1}{2}$$

Así que una cota superior u para $\hat{\alpha}$ en nuestro algoritmo sería $u \approx \frac{1}{2}$. Por otro lado, si $\hat{f}(1)$ es considerablemente más grande que $\hat{f}(0)$, entonces $\hat{\alpha}$ puede llegar a ser muy pequeña, lo que probablemente no quisiéramos que sucediera. Esto indica que posiblemente la aproximación cuadrática $m_{CVL}(\alpha)$ a $\hat{f}(\alpha)$ en esta región no es buena, así que, imponemos una cota inferior $l = \frac{1}{10}$ para $\hat{\alpha}$, lo que quiere decir que si en nuestro algoritmo $\hat{\alpha} \leq \frac{1}{10}$ tomaremos $\alpha_k = \frac{1}{10}$.

Ahora supóngase que $\hat{f}(\alpha_k) = f(x_k + \alpha_k p_k)$ sigue siendo mayor que $f(x_k) + \lambda \alpha_k \nabla^T f(x_k) p_k = \hat{f}(0) + \lambda \alpha_k \hat{f}'(0) p_k$, entonces necesitamos reducir aún más α_k . Contamos con la siguiente información acerca de $\hat{f}(\alpha)$: $\hat{f}(0)$, $\hat{f}'(0)$ y los dos últimos valores de $\hat{f}(\alpha)$, así que usaremos interpolación cúbica $m_{CVB}(\alpha)$ para aproximar $\hat{f}(\alpha)$ y posteriormente calcular el mínimo $\hat{\alpha}$ de $m_{CVB}(\alpha)$. Una idea geométrica de lo anterior, está representada en la siguiente figura.

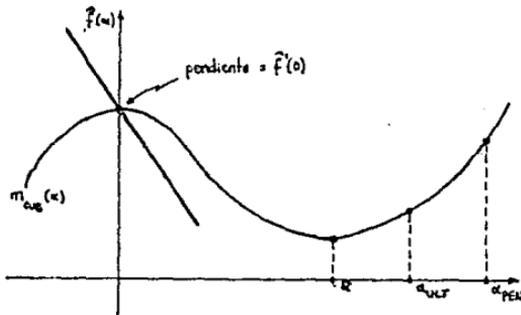


Fig. 14

El cálculo de $\hat{\alpha}$ es como sigue. Sean α_{UCL} y α_{PEN} los últimos valores de α . Entonces el modelo cúbico $m_{CVB}(\alpha)$ que interpola a $\hat{f}(\alpha)$ es

$$m_{CVB}(\alpha) = a\alpha^3 + b\alpha^2 + c\alpha + d$$

$$\begin{aligned}
 a &= \left(\frac{1}{\alpha_{ULT} - \alpha_{PEN}} \right) \left[\frac{1}{\alpha_{ULT}^2} (\hat{f}(\alpha_{ULT}) - \hat{f}(0) - \hat{f}'(0)\alpha_{ULT}) \right] \\
 &\quad - \left(\frac{1}{\alpha_{ULT} - \alpha_{PEN}} \right) \left[\frac{1}{\alpha_{PEN}^2} (\hat{f}(\alpha_{PEN}) - \hat{f}(0) - \hat{f}'(0)\alpha_{PEN}) \right] \\
 b &= \left(\frac{1}{\alpha_{ULT} - \alpha_{PEN}} \right) \left[-\frac{\alpha_{PEN}}{\alpha_{ULT}} (\hat{f}(\alpha_{ULT}) - \hat{f}(0) - \hat{f}'(0)\alpha_{ULT}) \right] \\
 &\quad + \left(\frac{1}{\alpha_{ULT} - \alpha_{PEN}} \right) \left[\frac{\alpha_{ULT}}{\alpha_{PEN}^2} (\hat{f}(\alpha_{PEN}) - \hat{f}(0) - \hat{f}'(0)\alpha_{PEN}) \right] \\
 c &= m'_{OUB}(0) = \hat{f}'(0) \\
 d &= m_{CUB}(0) = \hat{f}(0)
 \end{aligned}$$

cuyo mínimo es

$$\hat{\alpha} = \frac{-b + \sqrt{b^2 - 3a\hat{f}'(0)}}{3a}$$

requeriremos que $u = \frac{1}{2}$ sea una cota superior, que quiere decir, que si $\hat{\alpha} > \frac{1}{2}\alpha_{ULT}$ hacemos $\alpha_k = \frac{1}{2}\alpha_{ULT}$. También, puesto que $\hat{\alpha}$ puede llegar a ser una fracción muy pequeña de α_{ULT} , adoptamos la cota inferior $l = \frac{1}{10}$, es decir, si $\hat{\alpha} < \frac{1}{10}\alpha_{ULT}$, entonces tomamos $\alpha_k = \frac{1}{10}\alpha_{ULT}$. Se verifica con muchas operaciones y mediante las sustituciones apropiadas que $\hat{\alpha}$ nunca es imaginaria cuando $\lambda \leq \frac{1}{4}$. Finalmente el algoritmo nos queda como sigue:

Algoritmo "Búsqueda sobre una línea"

Dada $f: \mathbb{R}^n \rightarrow \mathbb{R}$ $\lambda = 10^{-4}$

Para $k = 0, 1, 2, 3, \dots$

Dada p_k = dirección descendente en x_k hacer:

$\alpha_k = 1$

$\alpha_{PEN} = \alpha_k$

(Almacenamiento de α_k en $\alpha_{PEN} = \alpha$ -penúltima).

Si

(I) $f(x_k + \alpha_k p_k) > f(x_k) + \lambda \alpha_k \nabla^T f(x_k) p_k$ hacer

(Un primer intento para calcular α_k adecuada)

- 1) $\alpha_k = -\frac{\nabla^T f(x_k) p_k}{2(f(x_k + p_k) - f(x_k)) - \nabla^T f(x_k) p_k}$
- 2) Si $\alpha_k \leq \frac{1}{10}$, entonces tomar $\alpha_k = \frac{1}{10}$
- 3) $\alpha_{ULT} = \alpha_k$ (Almacenamiento de α_k en $\alpha_{ULT} = \alpha$ última).
- 4) Si aún se sigue cumpliendo (I), entonces

Repetir

$$i) \hat{\alpha} = \frac{-b + \sqrt{b^2 - 3a \nabla^T f(x_k)}}{3a} \quad \text{donde}$$

$$i.1) a = \alpha_{DIF} \left(\frac{1}{\alpha_{ULT}^2} f_{ULT} - \frac{1}{\alpha_{PEN}^2} f_{PEN} \right)$$

$$i.2) b = \alpha_{DIF} \left(-\frac{\alpha_{PEN}}{\alpha_{ULT}} f_{ULT} + \frac{\alpha_{ULT}}{\alpha_{PEN}} f_{PEN} \right)$$

$$i.3) \alpha_{DIF} = \frac{1}{\alpha_{ULT} - \alpha_{PEN}}$$

$$i.4) f_{PEN} = f(x_k + \alpha_{PEN} p_k) - f(x_k) - \alpha_{PEN} \nabla^T f(x_k) p_k$$

$$i.5) f_{ULT} = f(x_k + \alpha_{ULT} p_k) - \alpha_{ULT} \nabla^T f(x_k) p_k - f(x_k)$$

$$ii) \text{ Si } \hat{\alpha} > \frac{\alpha_{ULT}}{10}, \text{ entonces } \hat{\alpha} = \frac{\alpha_{ULT}}{10}$$

$$iii) \text{ Si } \hat{\alpha} < \frac{\alpha_{ULT}}{10}, \text{ entonces } \hat{\alpha} = \frac{\alpha_{ULT}}{10}$$

$$iv) \alpha_{PEN} = \alpha_k$$

$$v) \alpha_k = \hat{\alpha}$$

Hasta que (I) no sea válido

(II) Calcular

$$x_{k+1} = x_k + \alpha_k p_k$$

4. Método de Newton con búsqueda lineal

Como se hizo notar en la primera sección de este capítulo, encontrar $x_* \in \mathbb{R}^n$ tal que $F(x_*) = 0$ con $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$ y $f \in C^1$ nos lleva a resolver el problema

$$\min_{x \in \mathbb{R}^n} f(x)$$

con

$$f: \mathbb{R}^n \rightarrow \mathbb{R} \quad f(x) = \frac{1}{2} \|F(x)\|_2^2$$

y además toda solución a $F(x) = 0$ es solución a $\min_{x \in \mathbb{R}^n} f(x)$. El inverso es falso (véase figura 2).

En este apartado trataremos de describir una estrategia global para resolver el problema $\min_{x \in \mathbb{R}^n} f(x)$ de tal forma que nos lleve a obtener en la medida de lo posible, una buena aproximación al punto x_* que satisfaga $F(x_*) = 0$.

Una pregunta interesante es la siguiente: ¿Cuáles son las direcciones de descenso del problema $\min_{x \in \mathbb{R}^n} f(x)$?

Como

$$\nabla f(x_k) = J^T(x_k)F(x_k),$$

entonces la dirección de descenso más rápida es

$$-\nabla f(x_k) = -J^T(x_k)F(x_k),$$

además la dirección de Newton

$$s_k = -J^{-1}(x_k)F(x_k),$$

también es una dirección de descenso puesto que

$$\nabla^T f(x_k)s_k = -F^T(x_k)J(x_k)J^{-1}(x_k)F(x_k) = -F^T(x_k)F(x_k) < 0$$

siempre y cuando $F(x_k) \neq 0$. Lo anterior es sorprendente, pero es geoméricamente razonable, ya que la dirección de Newton es solución de

$$M_k(x_k + s) = F(x_k) + J(x_k)s = 0$$

y también un mínimo de la función cuadrática

$$\begin{aligned} \hat{m}_k(x_k + s) &= \frac{1}{2}M_k^T(x_k + s)M_k(x_k + s) \\ &= \frac{1}{2}F^T(x_k)F(x_k) + (J^T(x_k)F(x_k))^T s \\ &\quad + \frac{1}{2}s^T(J^T(x_k)J(x_k))s \end{aligned}$$

debido a que $\hat{m}_k(x_k + s) \geq 0$ para todo s y $\hat{m}_k(x_k + s_k) = 0$. Por consiguiente, s_k es una dirección descendente para \hat{m}_k y también para f , ya que

$$\nabla \hat{m}_k(x_k) = \nabla f(x_k)$$

por lo tanto, el modelo $\hat{m}_k(x)$ tiene como mínimo a la dirección de Newton y como direcciones de descenso las de la función f .

Nótese que $\hat{m}_k(x)$ no es igual a la aproximación de Taylor cuadrática $m_k(x)$ alrededor del punto x_k de la función $f(x) = \frac{1}{2}F^T(x_k)F(x_k)$, puesto que

$$m_k(x_k + s) = f(x_k) + \nabla^T f(x_k)s + \frac{1}{2}s^T \nabla^2 f(x_k)s$$

y $\nabla^2 f(x_k) \neq J^T(x_k)J(x_k)$.

Luego, siempre y cuando $J^T(x_k)J(x_k)$ sea positiva definida, la aplicación del algoritmo "Búsqueda sobre una línea" a la función $f(x) = \|F(x)\|_2^2$ es directa.

Algoritmo "Newton con búsqueda sobre una línea"

Dada $F: \mathbb{R}^n \rightarrow \mathbb{R}^m$ de clase C^1 en $D \subset \mathbb{R}^n$, $x_0 \in D$

Para $k = 0, 1, 2, \dots$

Hacer

i) resolver $J(x_k)s_k = -F(x_k)$ para s_k

ii) $f(x_k) = \frac{1}{2}\|F(x_k)\|_2^2$

iii) determinar α_k que satisfaga

$$f(x_k + \alpha_k p_k) \leq f(x_k) + \lambda \alpha_k \nabla^T f(x_k) p_k$$

mediante el método "búsqueda sobre una línea"

iv) Calcular

$$x_{k+1} = x_k + \alpha_k s_k$$

Debido a las condiciones de Wolfe-Powell, por medio de este método obtenemos el mínimo global x_* de la función F valiéndose de la dirección de Newton $s_k = -J^{-1}(x_k)F(x_k)$ siempre que $J(x_k)$ sea no-singular y bien condicionada, lo cual no siempre se tiene garantizado, logrando convergencia cuadrática en una vecindad de x_* .

Ejemplo

Sea $F: \mathbb{R}^2 \rightarrow \mathbb{R}^2$

$$F(x) = \begin{pmatrix} x_1^2 + x_2^2 - 2 \\ e^{x_1-1} + x_2^2 - 2 \end{pmatrix}$$

que tiene solución $x_* = (1, 1)^T$. Tomemos como condición inicial $x_0 = (2, 0.5)^T$. Sea $f(x) = \frac{1}{2} F^T(x) F(x)$. Entonces

$$J(x_0) = \begin{pmatrix} 4 & 1 \\ e & 0.75 \end{pmatrix} \quad F(x_0) = \begin{pmatrix} 2.25 \\ 0.843 \end{pmatrix}$$

y

$$s_0 = -J^{-1}(x_0)F(x_0) \approx \begin{pmatrix} -3.00 \\ 9.74 \end{pmatrix}$$

Con "Búsqueda sobre una línea" se obtiene el primer punto

$$x_1 = x_0 + s_0 \approx \begin{pmatrix} -1.00 \\ 10.24 \end{pmatrix}$$

de donde $F(x_1) \approx \begin{pmatrix} 104 \\ 1071 \end{pmatrix}$, lo que nos indica que s_0 no es un buen paso. El algoritmo otorga el valor de $\alpha_1 = 0.1$, luego:

$$x_1 = x_0 + 0.1s_0 \approx \begin{pmatrix} 1.70 \\ 1.47 \end{pmatrix}, \quad F(x_1) \approx \begin{pmatrix} 3.06 \\ 3.21 \end{pmatrix}$$

lo que muestra que x_1 no es aún buena aproximación. Nuevamente nuestro algoritmo calcula una nueva reducción de α , dándonos $\alpha_2 = 0.05$, luego

$$x_1 = x_0 + 0.05s_0 \approx \begin{pmatrix} 1.85 \\ 0.987 \end{pmatrix}, \quad F(x_1) \approx \begin{pmatrix} 2.40 \\ 1.30 \end{pmatrix}$$

esta nueva aproximación sigue siendo mala. Una nueva reducción da $\alpha_3 \approx 0.016$, por tanto

$$x_1 = x_0 + 0.0116s_0 \approx \begin{pmatrix} 1.965 \\ 0.613 \end{pmatrix}, \quad F(x_1) \approx \begin{pmatrix} 2.238 \\ 0.856 \end{pmatrix}$$

Este punto sí es una buena aproximación, ya que

$$f(x_1) \approx 2.87 < 2.89 \approx f(x_0) + 10^{-4}(0.0116)\nabla^T f(x_0)s_0.$$

En la proxima iteración

$$s_1 = -J^{-1}(x_1)F(x_1) \approx \begin{pmatrix} -1.22 \\ 2.07 \end{pmatrix}$$

$$x_2 = x_1 + s_1 \approx \begin{pmatrix} 0.750 \\ 2.78 \end{pmatrix}, \quad F(x_2) \approx \begin{pmatrix} 5.77 \\ 18.13 \end{pmatrix}$$

que nuevamente no es una buena aproximación. La primera reducción nos otorga el valor de $\alpha_1 = 0.1$. Luego

$$x_3 = x_2 + 0.1s_1 \approx \begin{pmatrix} 1.84 \\ 0.820 \end{pmatrix}, \quad F(x_3) \approx \begin{pmatrix} 2.07 \\ 0.876 \end{pmatrix}$$

lo que es una buena aproximación, ya que

$$f(x_3) \approx 2.53 < 2.87 \approx f(x_2) + 10^{-4}(0.1)\nabla^T f(x_2)s_1.$$

Continuando

$$s_2 = -J^{-1}(x_3)F(x_3) \approx \begin{pmatrix} -0.0756 \\ 0.0437 \end{pmatrix}$$

$$x_4 = x_3 + s_2 \approx \begin{pmatrix} 1.088 \\ 1.257 \end{pmatrix}, \quad F(x_4) \approx \begin{pmatrix} 0.762 \\ 1.077 \end{pmatrix}$$

Así que este último paso es muy bueno. A partir de este punto el método de Newton converge cuadráticamente a $x_* = (1, 1)^T$.

El método de Newton con búsqueda sobre la línea mejora considerablemente el de Newton y nos permite usarlo como un método global, sin embargo, todavía persisten algunas complicaciones como cuando J es casi singular. Existen varias formas de salvar esta dificultad, pero no abundaremos más sobre el método de Newton, porque para nuestros intereses el Jacobiano es muy costoso y, por lo tanto, necesitamos de un método que no requiera su evaluación, por lo que presentaremos uno que está basado en una generalización del de la secante debido a Broyden [2].

5. Método de Broyden

Empecemos considerando el problema siguiente

Dada $f : \mathbb{R} \rightarrow \mathbb{R}$

encontrar $x_* \in \mathbb{R}$ tal que

$$f(x_*) = 0$$

Para $x_k \in \mathbb{R}$ Newton (Sección 3) nos proporciona una nueva aproximación x_{k+1} :

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

Pero ¿Qué sucede si no disponemos de $f'(x)$? Una primera solución, es aproximar f por la línea secante que une los puntos $(x_k, f(x_k))$ y $(x_k + h_k, f(x_k + h_k))$, donde se elige h_k de tal manera que x_k y $x_k + h_k$ sean cercanos, como lo muestra la siguiente figura.

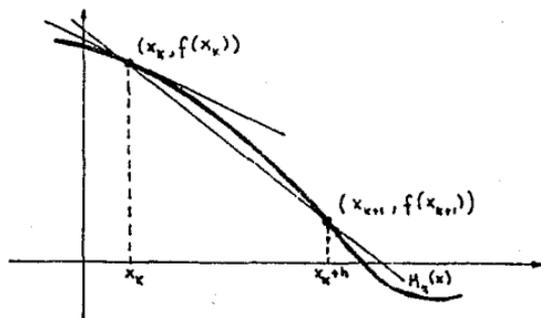


Fig. 15

y cuya pendiente es

$$a_k = \frac{f(x_k + h_k) - f(x_k)}{h_k}$$

de tal suerte, que podemos aproximar $f(x)$ por el modelo lineal

$$M_k(x) = f(x_k) + a_k(x - x_k)$$

al encontrar x_{k+1} , tal que $M_k(x_{k+1}) = 0$ nos lleva al método

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

¿Es eficiente este método? ¿Cómo se elige h_k ? Sabemos que $a_k \rightarrow f'(x_k)$ conforme $h_k \rightarrow 0$. Si h_k es escogido como un número pequeño, a_k recibe el nombre de aproximación por diferencias finitas a $f'(x_k)$.

Con la finalidad de evitar una evaluación extra de la función f ponemos $h_k = x_{k-1} - x_k$, así que

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

El algoritmo resultante es llamado método de la secante

Ahora consideremos el caso $n > 1$.

En una dimensión tenemos el modelo

$$M_k(x) = f(x_k) + a_k(x - x_k)$$

que cumple con $M_k(x_k) = f(x_k)$ para cualquier $a_k \in \mathbb{R}$ de donde obtenemos el método de Newton si $a_k = f'(x_k)$, y si

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

tenemos el método de la secante, así pues, el próximo punto por éste método lo obtenemos resolviendo $M_k(x_{k+1}) = 0$ de donde

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

En dimensión mayor el modelo análogo es

$$M_k(z) = F(x_k) + A_k(z - x_k)$$

que satisface $M_k(x_k) = F(x_k)$ para cualquier $A_k \in M_{n \times n}(\mathbb{R})$. Como vimos previamente, si $A_k = J(x_k)$ obtenemos el método de Newton. El problema surge nuevamente, como en el caso unidimensional, cuando no disponemos de $J(x_k)$ y de forma análoga, para obtener el método de la secante se requiere que $M_k(x_{k-1}) = F(x_{k-1})$, es decir

$$F(x_{k-1}) = F(x_k) + A_k(x_{k-1} - x_k)$$

o bien

$$A_k(x_k - x_{k-1}) = F(x_k) - F(x_{k-1})$$

que recibe el nombre de ecuación de la secante. Denotando $s_{k-1} = x_k - x_{k-1}$, $y_{k-1} = F(x_k) - F(x_{k-1})$, con lo que la ecuación de la secante la podemos reescribir de la forma

$$A_k s_{k-1} = y_{k-1}.$$

El problema de tratar de extender el método de la secante a dimensión mayor que uno, es que la matriz A_k no está completamente especificada por $A_k s_{k-1} = y_{k-1}$. En efecto, si $s_{k-1} \neq 0$ existe un subespacio afín de matrices $n \times n$ con dimensión $n(n-1)$, satisfaciendo cada uno de sus elementos $A_k s_{k-1} = y_{k-1}$. Así que, para poder construir una aproximación útil por el método de la secante, debemos de seleccionar adecuadamente una matriz perteneciente a dicho subespacio. Quizá la estrategia más obvia es requerir que el modelo

$$M_k(x) = F(x_k) + A_k(x - x_k)$$

interpole a F en $n-1$ puntos anteriores $x_{-1}, x_{-2}, \dots, x_{-(n-1)}$, es decir,

$$M_k(x_{-i}) = F(x_{-i})$$

o bien,

$$F(x_{-i}) = F(x_k) + A_k(x_{-i} - x_k)$$

lo que nos lleva a

$$A_k s_{-i} = y_{-i} \quad i = 1, 2, 3, \dots, n-1$$

donde $s_{-i} = x_{-i} - x_k$, $y_{-i} = F(x_{-i}) - F(x_k)$

Si $s_{k-1}, s_{-1}, \dots, s_{-(n-1)}$ son linealmente independientes entonces las n -ecuaciones

$$\begin{aligned}
 A_k s_{k-1} &= y_{k-1} \\
 A_k s_{k-1} &= y_{k-1} \\
 A_k s_{k-2} &= y_{k-2} \\
 &\vdots \\
 A_k s_{-(n-1)} &= y_{-(n-1)}
 \end{aligned}$$

determinan de manera única la matriz A_k .

Uno de los problemas principales que surgen de este método es que los vectores $s_{k-1}, s_{k-2}, \dots, s_{-(n-1)}$ tienden a ser linealmente dependientes conforme se calcula la matriz A_k , haciendo el cálculo de ésta un problema numérico mal condicionado, por lo que se hace uso de otra idea para calcular una mejor aproximación A_k de tal forma que satisfaga las condiciones previas, a saber:

Supongamos que tenemos una aproximación A_{k-1} en x_{k-1} y que queremos una aproximación A_k al Jacobiano en x_k , entonces lo que vamos a exigir a A_k es que

- 1) $A_k s_{k-1} = y_{k-1}$
- 2) $\|A_k - A_{k-1}\|$ sea lo más pequeña posible.

Para precisar lo anterior, observemos que la diferencia entre el nuevo modelo $M_k(x)$ y el viejo $M_{k-1}(x)$ es

$$\begin{aligned}
 M_k(x) - M_{k-1}(x) &= F(x_k) + A_k(x - x_k) - F(x_{k-1}) - A_{k-1}(x - x_{k-1}) \\
 &= F(x_k) - F(x_{k-1}) - A_k(x_k - x_{k-1}) + (A_k - A_{k-1})(x - x_{k-1}) \\
 &= (A_k - A_{k-1})(x - x_{k-1})
 \end{aligned}$$

con la última igualdad debido a que

$$A_k(x_k - x_{k-1}) = F(x_k) - F(x_{k-1})$$

Ahora, para cualquier $x \in \mathbb{R}^n$ expresemos

$$x - x_{k-1} = \alpha s_{k-1} + t \quad \text{con} \quad t^T s_{k-1} = 0$$

luego, el término que deseamos minimizar llega a ser

$$\begin{aligned}
 M_k(x) - M_{k-1}(x) &= (A_k - A_{k-1})(\alpha s_{k-1} + t) \\
 &= (A_k - A_{k-1})(\alpha s_{k-1} + t) \\
 &= \alpha(A_k - A_{k-1})s_{k-1} + (A_k - A_{k-1})t
 \end{aligned}$$

no tenemos control sobre el primer término del lado derecho ya que $(A_k - A_{k-1})s_{k-1} = y_{k-1} - A_{k-1}s_{k-1}$. Sin embargo, pidamos que el segundo término sea igual a cero para todo $x \in \mathbb{R}^n$, escogiendo A_k tal que

$$(A_k - A_{k-1})t = 0$$

para todo t ortogonal a s_{k-1} . Ésto requiere que $A_k - A_{k-1}$ sea una matriz de rango uno de la forma

$$A_k - A_{k-1} = us_{k-1}^T \quad u \in \mathbb{R}^n$$

entonces tenemos $(A_k - A_{k-1})s_{k-1} = y_{k-1} - A_{k-1}s_{k-1}$.

de donde

$$us_{k-1}^T s_{k-1} = y_{k-1} - A_{k-1}s_{k-1}$$

lo que implica que

$$u = \frac{y_{k-1} - A_{k-1}s_{k-1}}{s_{k-1}^T s_{k-1}}$$

por lo tanto de $A_k - A_{k-1} = us_{k-1}^T$ se deduce

$$A_k = A_{k-1} + \frac{y_{k-1} - A_{k-1}s_{k-1}}{s_{k-1}^T s_{k-1}} s_{k-1}^T$$

de donde

$$A_k s_{k-1} = A_{k-1} s_{k-1} + \frac{y_{k-1} - A_{k-1} s_{k-1}}{s_{k-1}^T s_{k-1}} s_{k-1}^T s_{k-1} = y_{k-1}$$

es decir, cumple con

$$A_k s_{k-1} = y_{k-1}$$

El siguiente resultado nos muestra que A_k es la matriz más próxima a A_{k-1} y que es consistente con $A_k s_{k-1} = y_{k-1}$ si el cambio $A_k - A_{k-1}$ es medido en la norma Frobenius. Pero antes veamos algunos preliminares. Denotemos

$$Q(y, s) = \{B \in M_{n \times n}(\mathbb{R}) : Bs = y\}$$

nótese que $Q(y, s)$ es un subespacio afín de $M_{n \times n}(\mathbb{R})$ ya que

$$Q(y, s) = B_0 + Q(0, s) \quad \text{con} \quad B_0 \in Q(y, s)$$

Lema

Sea $A \in M_{n \times n}(\mathbb{R})$, $s, y \in \mathbb{R}^n$, $s \neq 0$, entonces para cualquier norma $\|\cdot\|$ tal que

$$\|AB\| \leq \|A\| \|B\| \quad (1)$$

y

$$\left\| \frac{vy^T}{v^T v} \right\| = 1 \quad (2)$$

La solución al problema

$$\min_{B \in Q(y, s)} \|B - A\| \quad (3)$$

es

$$A_+ = A + \frac{(y - As)s^T}{s^T s} \quad (4)$$

En particular, (4) resuelve (3) cuando $\|\cdot\|$ es la norma l_2 y lo hace de manera única cuando es la norma de Frobenius.

Demostración:

Sea $B \in Q(y, s)$ entonces

$$\begin{aligned} \|A_+ - A\| &= \left\| \frac{(y - As)s^T}{s^T s} \right\| = \left\| \frac{(B - A)s s^T}{s^T s} \right\| \\ &\leq \|B - A\| \left\| \frac{s s^T}{s^T s} \right\| \leq \|B - A\| \end{aligned}$$

Si $\|\cdot\|$ es la norma l_2 , entonces

$$\|AB\|_2 \leq \|A\|_2 \|B\|_2 \leq \|A\|_2 \|B\|_F$$

porque $\|B\|_2 \leq \|B\|_F$.

$$\begin{aligned} \left\| \frac{vv^T}{v^T v} \right\|_F &= \frac{1}{v^T v} \|vv^T\|_F \\ &= \frac{1}{v^T v} \|v\|_2 \|v^T\|_2 \\ &= \frac{1}{v^T v} \|v\|_2^2 = 1 \end{aligned}$$

Además A_+ es la única solución a

$$\min_{B \in Q(y,s)} \|B - A\|_F$$

por ser la norma Frobenius estrictamente convexa y $Q(y, s)$ un subespacio afín convexo de $M_{n \times n}(\mathbb{R})$.

□

De lo anterior, podemos concluir que ya tenemos un método para cuando no nos es posible calcular el Jacobiano.

Algoritmo "Método de Broyden"

Dada $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$, $x_0 \in \mathbb{R}^n$, $A_0 \in M_{n \times n}(\mathbb{R})$

Para $k = 0, 1, 2, 3, \dots$

Resolver

$$A_k s_k = -F(x_k) \text{ para } s_k$$

$$x_{k+1} = x_k + s_k$$

$$y_k = F(x_{k+1}) - F(x_k)$$

$$A_{k+1} = A_k + \frac{(y_k - A_k s_k) s_k^T}{s_k^T s_k}$$

¿Cómo obtener una buena aproximación A_0 para $J(x_0)$? En nuestra implementación, usaremos una aproximación a $J(x_0)$ por diferencias.

6. Análisis de Convergencia del Método de Broyden.

En este apartado investigaremos la convergencia del algoritmo de Broyden. Mostraremos que si x_0 está suficientemente cercano a una raíz x_* de $F(x)$, $J(x_*)$ es

no-singular, y si A_0 es cercana a $J(x_0)$, entonces la sucesión generada $\{x_k\}$ converge superlinealmente a x_* . Es decir, que

$$\lim_{k \rightarrow \infty} \frac{\|x_{k+1} - x_k\|}{\|x_k - x_*\|} = 0$$

Lema

Sea $F: \mathbb{R}^n \rightarrow \mathbb{R}^m$ continuamente diferenciable en un conjunto abierto convexo $D \subset \mathbb{R}^n$, $x \in D$, J Lipschitz continua en x en D . Entonces para cualquier $x + p \in D$

$$\|F(x+p) - F(x) - J(x)p\| \leq \frac{\gamma}{2} \|p\|^2$$

donde γ es la constante de Lipschitz

Demostración:

$$\begin{aligned} F(x+p) - F(x) - J(x)p &= \int_0^1 J(x+tp)p dt - J(x)p \\ &= \int_0^1 (J(x+tp) - J(x))p dt \\ \|F(x+p) - F(x) - J(x)p\| &\leq \int_0^1 \|J(x+tp) - J(x)\| \|p\| dt \\ &\leq \int_0^1 \gamma \|tp\| \|p\| dt \\ &= \gamma \|p\|^2 \int_0^1 t dt = \frac{\gamma}{2} \|p\|^2 \end{aligned}$$

□

Ahora bien, si $F(x_*) = 0$, entonces de la iteración

$$x_{k+1} = x_k - A_k^{-1} F(x_k)$$

tenemos

$$x_{k+1} - x_* = x_k - x_* - A_k^{-1} (F(x_k) - F(x_*))$$

o bien

$$A_k(x_{k+1} - x_*) = A_k(x_k - x_*) - F(x_k) + F(x_*)$$

haciendo $e_k := x_k - x_*$, sumando y restando $J(x_*)e_k$ en el lado derecho de la anterior ecuación obtenemos

$$A_k e_{k+1} = [-F(x_k) + F(x_*) + J(x_*)e_k] + (A_k - J(x_*))e_k$$

por el lema anterior con $x = x_*$,

$$\| -F(x_k) + F(x_*) + J(x_*)e_k \| = O\|e_k\|^2$$

Así que, la clave para el análisis de la convergencia del método de Broyden se referirá al término $(A_k - J(x_*))e_k$. Mostraremos que

$$\lim_{k \rightarrow \infty} \frac{\|(A_k - J(x_*))e_k\|}{\|e_k\|} = 0$$

que es lo que en realidad deseamos de la aproximación al Jacobiano y ello implica que el paso de la secante $-A_k^{-1}F(x_k)$, converja al paso de Newton $-J^{-1}(x_k)F(x_k)$ en magnitud y dirección.

Pero antes probemos un resultado

Lema

Sea $x_k \in \mathbb{R}^n$ $k = 0, 1, 2, \dots$ Si $\{x_k\}$ converge superlinealmente a $x_* \in \mathbb{R}^n$ en alguna norma $\|\cdot\|$, entonces

$$\lim_{k \rightarrow \infty} \frac{\|s_k\|}{\|e_k\|} = 1$$

Demostración:

El resultado se sigue de la siguiente figura:

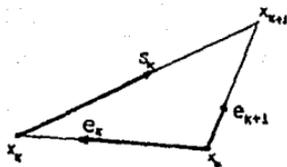


Fig. 16

claramente, si

$$\lim_{k \rightarrow \infty} \frac{\|e_{k+1}\|}{\|e_k\|} = 0 \implies \lim_{k \rightarrow \infty} \frac{\|s_k\|}{\|e_k\|} = 1$$

ya que

$$\lim_{k \rightarrow \infty} \left| \frac{\|s_k\|}{\|e_k\|} - 1 \right| = \lim_{k \rightarrow \infty} \left| \frac{\|s_k\| - \|e_k\|}{\|e_k\|} \right| \leq \lim_{k \rightarrow \infty} \left| \frac{\|s_k + e_k\|}{\|e_k\|} \right| = \lim_{k \rightarrow \infty} \frac{\|e_{k+1}\|}{\|e_k\|} = 0$$

donde la última igualdad es la definición de convergencia superlineal cuando $e_k \neq 0$ para toda k .

□

El siguiente teorema nos proporciona condiciones necesarias y suficientes para la convergencia superlineal de un método cuasi-newton. Pero antes necesitamos dos resultados previos.

Lema

Sea $F: \mathbb{R}^n \rightarrow \mathbb{R}^m$ continuamente diferenciable en un conjunto abierto convexo $D \subset \mathbb{R}^n$, $x \in D$, J Lipschitz continua en x en la vecindad D . Entonces para cualquier $u, v \in D$.

$$(I) \quad \|F(v) - F(u) - J(x)(v - u)\| \leq \gamma \frac{\|v - x\| + \|u - x\|}{2} \|v - u\|.$$

donde γ es la constante de Lipschitz.

Supóngase además que $J^{-1}(x)$ existe. Entonces existen $\epsilon > 0$, $0 < \alpha < \beta$, tal que

$$(II) \quad \alpha \|v - u\| \leq \|F(v) - F(u)\| \leq \beta \|v - u\| \quad \forall u, v \in D \text{ para los cuales } \max\{\|v - x\|, \|u - x\|\} \leq \epsilon$$

Demostración de (I):

Se tiene

$$F(v) - F(u) = F(u + v - u) - F(u) = \int_0^1 J(u + t(v - u)) \cdot (v - u) dt$$

luego

$$\|F(v) - F(u) - J(x)(v - u)\| \leq \int_0^1 \|J(u + t(v - u)) - J(x)\| \cdot (v - u) dt.$$

Usando el hecho de que J es Lipschitz en x , escribimos $u + t(v - u) = x + u + t(v - u) - x$ para así tener que

$$\begin{aligned} \int_0^1 \|J(u + t(v - u)) - J(x)\| \|v - u\| dt &= \|v - u\| \int_0^1 \|J(x + u + t(v - u) - x) - J(x)\| dt \\ &\leq \|v - u\| \gamma \int_0^1 \|u + t(v - u) - x\| dt \end{aligned}$$

Ahora tenemos que $v - u = v - u + x - x$ de donde

$$u + t(v - u) - x = u + t(v - x + x - u) - x = (x - u)(t - 1) + (v - x)t$$

sustituyendo y usando la desigualdad del triángulo:

$$\begin{aligned} \|v - u\| \gamma \int_0^1 \|u + t(v - u) - x\| dt &= \|v - u\| \gamma \int_0^1 \|(x - u)(t - 1) + (v - x)t\| dt \\ &\leq \|v - u\| \gamma \left[\|v - x\| \int_0^1 t dt \right] = \gamma \frac{\|v - u\|}{2} (\|u - x\| + \|v - x\|) \end{aligned}$$

Por lo tanto

$$\|F(v) - F(u) - J(x) \cdot (v - u)\| \leq \frac{\gamma}{2} \|v - u\| (\|u - x\| + \|v - x\|)$$

□

Demostración de (II):

Usando la desigualdad del triángulo y el resultado anterior, tenemos

$$\begin{aligned} \|F(v) - F(u)\| &\leq \|J(x) \cdot (v - u)\| + \|F(v) - F(u) - J(x) \cdot (v - u)\| \\ &\leq \left[\|J(x)\| + \frac{\gamma}{2} (\|v - x\| + \|u - x\|) \right] \|v - u\| \\ &\leq (\|J(x)\| + \gamma\epsilon) \|v - u\| \end{aligned}$$

que prueba la segunda parte de (II) con $\beta = \|J(x)\| + \gamma\epsilon$. Similarmente

$$\begin{aligned} \|F(v) - F(u)\| &\geq \|J(x) \cdot (v - u)\| - \|F(v) - F(u) - J(x) \cdot (v - u)\| \\ &\geq \left[\left(\frac{1}{\|J^{-1}(x)\|} \right) - \frac{\gamma}{2} (\|v - x\| + \|u - x\|) \right] \|v - u\| \\ &\geq \left[\frac{1}{\|J^{-1}(x)\|} - \gamma\epsilon \right] \|v - u\| \end{aligned}$$

Así, si $\epsilon < \frac{1}{\|J^{-1}(x)\| \gamma}$ la primera parte de (II) se cumple con

$$\alpha = \left(\frac{1}{\|J^{-1}(x)\|} \right) - \gamma\epsilon > 0$$

lo que prueba (II).

□

Ahora si tenemos el resultado esperado.

Teorema

Sea $D \subset \mathbb{R}^n$ un conjunto abierto convexo, $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$, $J(x) \in \text{Lip}_\gamma(D)$, $x_* \in D$ con $J(x_*)$ no singular. Sea $\{A_k\}$ una sucesión de matrices no singulares $n \times n$. Sea $x_0 \in D$ y la sucesión $x_{k+1} = x_k - A_k^{-1}F(x_k)$ tal que x_k pertenece a D y $x_k \neq x_* \forall k$. Entonces $\{x_k\}$ converge superlinealmente a x_* con alguna norma $\|\cdot\|$, y

$$F(x_*) = 0 \iff \lim_{k \rightarrow \infty} \frac{\|(A_k - J(x_*))s_k\|}{\|s_k\|} = 0$$

donde $s_k = x_{k+1} - x_k$.

Demostración.

Sea $J_* = J(x_*)$, $e_k = x_k - x_*$. Supongamos primero que

$$\lim_{k \rightarrow \infty} \frac{\|(A_k - J(x_*))s_k\|}{\|s_k\|} = 0$$

Por demostrar que $F(x_*) = 0$ y que $\{x_k\} \rightarrow x_*$ superlinealmente. Tenemos

$$\begin{aligned} 0 &= A_k(x_{k+1} - x_k) + F(x_k) \\ &= A_k s_k + F(x_k) \\ &= (A_k - J_*)s_k + F(x_k) + J_* s_k \end{aligned}$$

Así que

$$\begin{aligned} -F(x_{k+1}) &= (A_k - J_*)s_k + \{-F(x_{k+1}) + F(x_k) + J_* s_k\} \\ \frac{\|F(x_{k+1})\|}{\|s_k\|} &\leq \frac{\|(A_k - J_*)s_k\|}{\|s_k\|} + \frac{\| -F(x_{k+1}) + F(x_k) + J_* s_k \|}{\|s_k\|} \\ &\leq \frac{\|(A_k - J_*)s_k\|}{\|s_k\|} + \frac{\gamma}{2} (\|e_k\| + \|e_{k+1}\|) \end{aligned}$$

la última desigualdad debido a (I) que recién se probó. Usando el hecho

$$\lim_{k \rightarrow \infty} \|e_k\| = 0$$

la hipótesis y la última desigualdad

$$\lim_{k \rightarrow \infty} \frac{\|F(x_{k+1})\|}{\|s_k\|} = 0$$

ya que

$$\lim_{k \rightarrow \infty} \|s_k\| = 0$$

ésto implica

$$F(x_*) = \lim_{k \rightarrow \infty} F(x_k) = 0$$

De (II) existen $\alpha > 0$, $k_0 \geq 0$ tal que

$$(**) \quad \|F(x_{k+1})\| = \|F(x_{k+1}) - F(x_*)\| \geq \alpha \|e_{k+1}\|$$

para todo $k \geq k_0$.

De (*) y (**) obtenemos

$$\begin{aligned} 0 &= \lim_{k \rightarrow \infty} \frac{\|F(x_{k+1})\|}{\|s_k\|} \geq \lim_{k \rightarrow \infty} \alpha \frac{\|e_{k+1}\|}{\|s_k\|} \\ &\geq \lim_{k \rightarrow \infty} \alpha \frac{\|e_{k+1}\|}{\|e_k\| + \|e_{k+1}\|} = \lim_{k \rightarrow \infty} \frac{\alpha \frac{\|e_{k+1}\|}{\|e_k\|}}{1 + \frac{\|e_{k+1}\|}{\|e_k\|}} \end{aligned}$$

lo que implica que

$$\lim_{k \rightarrow \infty} \frac{\|e_{k+1}\|}{\|e_k\|} = 0$$

que completa la prueba de la convergencia superlineal.

Ahora supóngase que $\{x_k\}$ converge superlinealmente a x_* y $F(x_*) = 0$. Probemos que

$$\lim_{k \rightarrow \infty} \frac{\|(A_k - J(x_*))s_k\|}{\|s_k\|} = 0$$

De (II) existen $\beta > 0$, $k_0 \leq 0$ tal que

$$\|F(x_{k+1}) - F(x_*)\| = \|F(x_{k+1})\| \leq \beta \|x_{k+1} - x_*\| = \beta \|e_{k+1}\|$$

$\forall k \geq k_0$.

y la convergencia superlineal implica

$$\begin{aligned}
0 &= \lim_{k \rightarrow \infty} \frac{\|e_{k+1}\|}{\|e_k\|} \geq \lim_{k \rightarrow \infty} \frac{\|F(x_{k+1})\|}{\beta \|e_k\|} \\
&= \lim_{k \rightarrow \infty} \frac{1}{\beta} \frac{\|F(x_{k+1})\|}{\|s_k\|} \frac{\|s_k\|}{\|e_k\|}
\end{aligned}$$

Como vimos en (*)

$$\lim_{k \rightarrow \infty} \frac{\|s_k\|}{\|e_k\|} = 1$$

lo que nos lleva a

$$\lim_{k \rightarrow \infty} \frac{\|F(x_{k+1})\|}{\|s_k\|} = 0$$

luego de

$$-F(x_{k+1}) = (A_k - J_0)s_k + \{-F(x_{k+1}) + F(x_k) + J_0 s_k\}$$

y de (I)

$$\begin{aligned}
\frac{\|(A_k - J_0)s_k\|}{\|s_k\|} &\leq \frac{\|F(x_{k+1})\|}{\|s_k\|} + \frac{\|-F(x_{k+1}) + F(x_k) + J_0 s_k\|}{\|s_k\|} \\
&\leq \frac{\|F(x_{k+1})\|}{\|s_k\|} + \frac{\gamma(\|e_{k+1}\| + \|e_k\|)(\|s_k\|)}{2\|s_k\|} \\
&= \frac{\|F(x_{k+1})\|}{\|s_k\|} + \frac{\gamma}{2}(\|e_{k+1}\| + \|e_k\|)
\end{aligned}$$

luego

$$\begin{aligned}
\lim_{k \rightarrow \infty} \frac{\|(A_k - J_0)s_k\|}{\|s_k\|} &\leq \lim_{k \rightarrow \infty} \left[\frac{\|F(x_{k+1})\|}{\|s_k\|} + \frac{\gamma}{2}(\|e_{k+1}\| + \|e_k\|) \right] \\
&= \lim_{k \rightarrow \infty} \frac{\|F(x_{k+1})\|}{\|s_k\|} + \frac{\gamma}{2} \left[\lim_{k \rightarrow \infty} \|e_{k+1}\| + \lim_{k \rightarrow \infty} \|e_k\| \right] = 0
\end{aligned}$$

Por lo tanto

$$\lim_{k \rightarrow \infty} \frac{\|(A_k - J(x_0))s_k\|}{\|s_k\|} = 0$$

Observación. Puesto que $J(x)$ es Lipschitz continua, el resultado probado se sigue cumpliendo si se reemplaza

$$\lim_{k \rightarrow \infty} \frac{\|(A_k - J(x_*))s_k\|}{\|s_k\|} = 0$$

por

$$\lim_{k \rightarrow \infty} \frac{\|(A_k - J(x_k))s_k\|}{\|s_k\|} = 0$$

lo cual tiene una interesante interpretación:

Como $s_k = -A_k^{-1}F(x_k)$ la condición

$$\lim_{k \rightarrow \infty} \frac{\|(A_k - J(x_k))s_k\|}{\|s_k\|} = 0$$

es equivalente a

$$\lim_{k \rightarrow \infty} \frac{\|J(x_k)(s_k^n - s_k)\|}{\|s_k\|} = 0$$

donde $s_k^n = -J^{-1}(x_k)F(x_k)$ es la dirección de Newton en x_k , ya que

$$\begin{aligned} J(x_k)(s_k^n - s_k) &= J(x_k)[-J^{-1}(x_k)F(x_k) + A_k^{-1}F(x_k)] \\ &= F(x_k) + J(x_k)A_k^{-1}F(x_k) \\ &= (A_k + J(x_k))A_k^{-1}F(x_k) \\ &= (A_k + J(x_k))s_k \end{aligned}$$

Así que la condición suficiente y necesaria para la convergencia superlineal del método de la secante es que la dirección de la secante s_k converja en magnitud y dirección a la dirección de Newton s_k^n .

La dificultad fuerte que se nos presenta al tratar de localizar el mínimo global x_* de F usando Newton es que, en este caso, la dirección

$$s_k = -J^{-1}(x_k)F(x_k)$$

es en efecto de descenso, pero no fácil de calcularla ya que, como se mencionó anteriormente, $J(x)$ puede ser singular o mal condicionada. Al estar por el método de Broyden, para que las condiciones de Wolfe-Powell nos garanticen convergencia es necesario que la dirección

$$s_k = -A^{-1}(x_k)F(x_k)$$

sea de descenso para la función $f(x) = \frac{1}{2}\|F(x)\|_2^2$, lo que no siempre es cierto, por lo que se tiene que tomar alguna precaución en este caso. Para ello calculemos una aproximación \hat{J}_k del Jacobiano por diferencias finitas y tomar como dirección

$$s_k = -\hat{J}^{-1}(x_k)F(x_k)$$

de esta forma siempre obtenemos una dirección de descenso y entonces las condiciones de Wolfe-Powell nos garantizan convergencia global obteniendo finalmente la convergencia superlineal del método de Broyden.

Algoritmo "Broyden con búsqueda sobre una línea"

Dada $F: \mathbb{R}^n \rightarrow \mathbb{R}^n$, $x_0 \in \mathbb{R}^n$, $A_0 \in M_{n \times n}(\mathbb{R})$, $x_0 \in \mathbb{R}^n$

Para $k = 0, 1, 2, 3, \dots$

Resolver

i) $A_k p_k = -F(x_k)$ para p_k

ii) $f(x_k) = \frac{1}{2}\|F(x_k)\|_2^2$

iii) determinar α_k que satisfaga

$$f(x_k + \alpha_k p_k) \leq f(x_k) + \lambda \alpha_k \nabla^T f(x_k) p_k$$

mediante el método "búsqueda sobre una línea"

iv) $x_{k+1} = x_k + \alpha_k p_k$

v) $y_k = F(x_{k+1}) - F(x_k)$

vi) $A_{k+1} = A_k + \frac{(y_k - A_k p_k) p_k^T}{p_k^T p_k}$

Ejemplo

Sea $F: \mathbb{R}^2 \rightarrow \mathbb{R}^2$

$$F(x) = \begin{pmatrix} x_1^2 + x_2^2 - 2 \\ e^{x_1-1} + x_2^3 - 2 \end{pmatrix}$$

que tiene una raíz en $x_* = (1, 1)^T$. Sea $x_0 = (1.5, 2)^T$. $A_0 = J(x_0)$ Los métodos de Newton y Broyden, ambos con búsqueda sobre una línea, nos dan

Broyden:

$$\begin{array}{ll} x_0 = \begin{pmatrix} 1.5000 \\ 2.0000 \end{pmatrix} & x_1 = \begin{pmatrix} 0.80606 \\ 1.45794 \end{pmatrix} \\ x_2 = \begin{pmatrix} 0.74107 \\ 1.27706 \end{pmatrix} & x_3 = \begin{pmatrix} 0.80227 \\ 1.15990 \end{pmatrix} \\ x_4 = \begin{pmatrix} 0.92946 \\ 1.07040 \end{pmatrix} & x_5 = \begin{pmatrix} 1.00402 \\ 1.00960 \end{pmatrix} \\ x_6 = \begin{pmatrix} 1.00308 \\ 0.99922 \end{pmatrix} & x_7 = \begin{pmatrix} 1.00054 \\ 0.99968 \end{pmatrix} \\ x_8 = \begin{pmatrix} 0.99999 \\ 1.00000 \end{pmatrix} & x_9 = \begin{pmatrix} 1.00000 \\ 1.00000 \end{pmatrix} \end{array}$$

Newton:

$$\begin{array}{ll} x_0 = \begin{pmatrix} 1.5000 \\ 2.0000 \end{pmatrix} & x_1 = \begin{pmatrix} 0.80606 \\ 1.45794 \end{pmatrix} \\ x_2 = \begin{pmatrix} 0.89011 \\ 1.14557 \end{pmatrix} & x_3 = \begin{pmatrix} 0.99158 \\ 1.02105 \end{pmatrix} \\ x_4 = \begin{pmatrix} 0.99970 \\ 1.00053 \end{pmatrix} & x_5 = \begin{pmatrix} 0.99999 \\ 1.00000 \end{pmatrix} \\ x_6 = \begin{pmatrix} 1.00000 \\ 1.00000 \end{pmatrix} & \end{array}$$

CAPÍTULO III

ELECCIÓN DE UN MÉTODO NUMÉRICO PARA RESOLVER ECUACIONES DIFERENCIALES ORDINARIAS

En este capítulo presentaremos las ideas básicas de los métodos de un-paso, que son los métodos llamados del tipo Runge-Kutta. El método que escogimos para resolver las ecuaciones diferenciales es este tipo y se debe a England [1,6,7] su creación. Parece ser uno de los métodos más eficaces, como se puede ver en el estudio de Shampine-Watts [7].

Los métodos numéricos para la solución de ecuaciones diferenciales o dinámicas, se pueden dividir de manera natural en dos tipos: los que usan un sólo valor de la solución y las que usan varios para comenzar en cada paso.

1. Métodos de Runge-Kutta

De los métodos más sencillos de entender e implementar numéricamente para resolver el problema de valor inicial

$$y' = f(x, y) \quad y(x_0) = y_0 \quad (1.1)$$

es el método de Euler que viene dado por

$$\begin{aligned} y_{n+1} &= y_n + h_n f(x_n, y_n) & n &= 0, 1, 2, \dots \\ h_n &= x_{n+1} - x_n \end{aligned}$$

Tenemos una cota para el error global que está dado por Ch donde C es una constante dependiendo del problema y h el máximo paso tomado. Cuando requerimos mucha exactitud, este método ya no es tan útil, por ejemplo, si uno necesitara 6 cifras decimales de exactitud, se tendrían que tomar un millón de pasos.

Se conocen métodos muchos más exactos cuando f es independiente de y , es decir, en el problema

$$y' = f(x) \quad y(x_0) = y_0$$

cuya solución viene dada por

$$y(x) = y_0 + \int_{x_0}^x f(x) dx$$

Como un ejemplo consideremos la regla del punto medio

$$y(x_0 + h_0) \approx y_1 = y_0 + h_0 f\left(x_0 + \frac{h_0}{2}\right)$$

$$y(x_1 + h_1) \approx y_2 = y_1 + h_1 f\left(x_1 + \frac{h_1}{2}\right)$$

⋮

$$y(x) \approx y = y_{n-1} + h_{n-1} f\left(x_{n-1} + \frac{h_{n-1}}{2}\right)$$

donde $h_i = x_{i+1} - x_i$ y $x_0, x_1, \dots, x_{n-1}, x_n = x$ es una subdivisión del intervalo de integración. Se verifica fácilmente que su error global $|y(x) - y|$ está acotada por Ch^2 , de tal suerte que si deseáramos precisión de 6 cifras decimales, se necesitarían aproximadamente mil pasos, es decir, que este método es mil veces más rápido que el de Euler. Por consiguiente, sería deseable poder extender este método para poder resolver el problema (1.1).

El primer punto con $h = h_0$ sería

$$y(x_0 + h_0) \approx y_1 = y_0 + h f\left(x_0 + \frac{h_0}{2}, y\left(x_0 + \frac{h_0}{2}\right)\right)$$

pero ¿Cuál sería el valor de $y\left(x_0 + \frac{h_0}{2}\right)$? Aplicando el método de Euler con paso $\frac{h_0}{2}$ para obtener y_1 llegamos a:

$$k_1 = f(x_0, y_0)$$

$$k_2 = f\left(x_0 + \frac{h_0}{2}, y_0 + \frac{h_0}{2} k_1\right)$$

$$y_1 = y_0 + h k_2$$

Esta forma de calcular y_1 usando un paso de Euler nos acerca la lentitud inherente a dicho método, debido a que h multiplica el coeficiente k_2 y por consiguiente el error es menos importante. Para ser más precisos, calculemos y_1 por su expansión de Taylor como función de h

$$\begin{aligned}
 y_1 &= y_0 + hf \left(x_0 + \frac{h_0}{2}, y_0 + \frac{h_0}{2} f_0 \right) \\
 &= y_0 + hf(x_0, y_0) + \frac{h^2}{2} (f_x + f_y f)(x_0, y_0) \\
 &\quad + \frac{h^3}{3} (f_{xx} + 2f_{xy}f + f_{yy}f^2)(x_0, y_0) + \dots
 \end{aligned}$$

Haciendo lo mismo con $y(x_0 + h)$

$$\begin{aligned}
 y(x_0 + h) &= y_0 + hf(x_0, y_0) + \frac{h^2}{2} (f_x + f_y f)(x_0, y_0) \\
 &\quad + \frac{h^3}{6} (f_{xx} + 2f_{xy}f + f_{yy}f^2 + f_y f_x + f_y^2 f)(x_0, y_0) + \dots
 \end{aligned}$$

Restando estas dos cantidades obtenemos el error en el primer paso

$$y(x_0 + h) - y_1 = \frac{h^3}{24} (f_{xx} + 2f_{xy}f + f_{yy}f^2 + 4(f_y f_x + f_y^2 f))(x_0, y_0) + \dots$$

Si todas las derivadas de segundo orden de f son acotadas, obtenemos

$$\|y(x_0 + h) - y_1\| \leq Kh^3$$

Luego para obtener el error cometido en el punto final x , procedemos de igual forma en los intervalos

$$(x_0, x_1), (x_1, x_2), \dots, (x_{n-1}, x)$$

encontrando que el error está acotado por Ch^3 .

Para obtener más precisión, necesitamos encontrar mejores métodos.

2. Formulación general de los métodos de Runge-Kutta

Runge (1895) y Heun (1900) construyeron métodos que incluían uno o dos pasos adicionales de Euler para obtener soluciones de (1.1). Fue precisamente Kutta (1901) quien formuló el esquema general del ahora llamado método de Runge-Kutta.

Definición

Sea $s \in \mathbb{N}$ y $a_{21}, a_{31}, \dots, a_{s1}, a_{22}, \dots, a_{s(s-1)}, b_1, \dots, b_s, c_1, \dots, c_s \in \mathbb{R}$. Entonces el método

$$\begin{aligned}k_1 &= f(x_0, y_0) \\k_2 &= f(x_0 + c_2 h, y_0 + h a_{21} k_1) \\k_3 &= f(x_0 + c_3 h, y_0 + h(a_{31} k_1 + a_{32} k_2)) \\&\vdots \\k_s &= f(x_0 + c_s h, y_0 + h(a_{s1} k_1 + \dots + a_{s(s-1)} k_{s-1})) \\y_1 &= y_0 + h(b_1 k_1 + \dots + b_s k_s)\end{aligned}$$

es llamado método de Runge-Kutta explícito de s -evaluaciones para el problema

$$y' = f(x) \quad y(x_0) = y_0$$

Usualmente los coeficientes c_i satisfacen las condiciones

$$c_2 = a_{21}, \quad c_3 = a_{31} + a_{32}, \dots, \quad c_s = a_{s1} + \dots + a_{s(s-1)}$$

o bien

$$c_i = \sum_{j=1}^{i-1} a_{ij}$$

Estas condiciones fueron supuestas por Kutta sin ningún comentario y expresó que todos los puntos donde f es evaluada son aproximaciones de primer orden a la solución, además de que también simplifican la derivación de las condiciones de orden para métodos de orden alto. Sin embargo, para ordenes bajos, dichas condiciones no son necesarias.

Definición

Un método de Runge-Kutta tiene orden p , si

$$\|y(x_0 + h) - y_1\| \leq K h^{p+1}$$

es decir, si las series de Taylor para la solución exacta $y(x_0 + h)$ y para y_1 coinciden hasta el término h^p .

Es costumbre simbolizar el método de Runge-Kutta por la siguiente tabla

0					
c_2	a_{21}				
c_3	a_{31}	a_{32}			
\vdots	\vdots	\vdots			
c_s	a_{s1}	a_{s2}	\dots	$a_{s(s-1)}$	
	b_1	b_2	\dots	b_{s-1}	b_s

Ejemplos

El método de Runge visto anteriormente, así como los métodos de Runge y Heun de orden 3 están dados de la siguiente manera

0	
$\frac{1}{2}$	$\frac{1}{2}$
	0 1

0				
$\frac{1}{3}$	$\frac{1}{3}$			
1	0	1		
1	0	0	1	
	$\frac{1}{6}$	$\frac{2}{3}$	0	$\frac{1}{6}$

$$\begin{aligned}
 k_1 &= f(x_0, y_0) \\
 k_2 &= f\left(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}hk_1\right) \\
 y_1 &= y_0 + hk_2
 \end{aligned}$$

$$\begin{aligned}
 k_1 &= f(x_0, y_0) \\
 k_2 &= f\left(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}hk_1\right) \\
 k_3 &= f(x_0 + h, y_0 + hk_2) \\
 k_4 &= f\left(x_0 + h, y_0 + hk_3\right) \\
 y_1 &= y_0 + h\left(\frac{1}{6}k_1 + \frac{2}{3}k_2 + \frac{1}{6}k_4\right)
 \end{aligned}$$

0	
$\frac{1}{3}$	$\frac{1}{3}$
$\frac{2}{3}$	0 $\frac{2}{3}$
$\frac{1}{4}$	0

$$k_1 = f(x_0, y_0)$$

$$k_2 = f\left(x_0 + \frac{1}{3}h, y_0 + \frac{1}{3}hk_1\right)$$

$$k_3 = f\left(x_0 + \frac{2}{3}h, y_0 + \frac{2}{3}hk_2\right)$$

$$y_1 = y_0 + \frac{1}{4}hk_1$$

Ahora quisiéramos determinar los coeficientes del método de Runge-Kutta cuando $s = 4$ de tal suerte que también obtengamos orden igual a 4. Lo que debemos de hacer es lo siguiente. Calcular las derivadas de

$$y_1 = y_0 + h(b_1k_1 + \dots + b_4k_4)$$

en $h = 0$ y compararlas con las de la solución verdadera para ordenes 1,2,3 y 4 usando las condiciones $c_i = \sum_j a_{ij}$ con lo cual obtenemos:

(2.1)

$$\sum_i b_i = b_1 + b_2 + b_3 + b_4 = 1 \quad (a)$$

$$\sum_i b_i c_i = b_2 c_2 + b_3 c_3 + b_4 c_4 = \frac{1}{2} \quad (b)$$

$$\sum_i b_i c_i^2 = b_2 c_2^2 + b_3 c_3^2 + b_4 c_4^2 = \frac{1}{3} \quad (c)$$

$$\sum_{i,j} b_i a_{ij} c_j = b_3 a_{31} c_1 + b_4 (a_{41} c_1 + a_{42} c_2) = \frac{1}{6} \quad (d)$$

$$\sum_i b_i c_i^3 = b_2 c_2^3 + b_3 c_3^3 + b_4 c_4^3 = \frac{1}{4} \quad (e)$$

$$\sum_{i,j} b_i c_i a_{ij} c_j = b_3 c_3 a_{32} c_2 + b_4 c_4 (a_{41} c_1 + a_{42} c_2) = \frac{1}{8} \quad (f)$$

$$\sum_{i,j} b_i a_{ij} c_j^2 = b_3 a_{32} c_2^2 + b_4 (a_{42} c_2^2 + a_{43} c_3^2) = \frac{1}{12} \quad (g)$$

$$\sum_{i,j} b_i a_{ij} a_{jk} c_k = b_4 a_{43} a_{32} c_2 = \frac{1}{24} \quad (h)$$

Kutta dio la solución general del anterior sistema sin ningún comentario. Nosotros daremos la idea de solución dada en [6].

Lema 1

Si

$$\sum b_i a_{ij} = b_j(1 - c_j) \quad j = 1, 2, 3, 4 \quad (2.2)$$

entonces, las ecuaciones (d), (g) y (h) se deducen de las restantes.

Lema 2

Para $s = 4$ el sistema (2.1) y las condiciones sobre c_i implican (2.2).

De los anteriores resultados obtenemos

Teorema

Suponiendo las condiciones

$$c_i = \sum_{j=1}^{i-1} a_{ij}$$

el sistema de ecuaciones (2.1) es equivalente a

$$\begin{aligned} b_1 + b_2 + b_3 + b_4 &= 1 \\ b_2 c_2 + b_3 b_3 + b_4 b_4 &= \frac{1}{2} \\ b_2 c_2^2 + b_3 c_3^2 + b_4 c_4^2 &= \frac{1}{3} \\ b_2 c_2^3 + b_3 c_3^3 + b_4 c_4^3 &= \frac{1}{4} \\ b_3 c_3 a_{32} c_2 + b_4 c_4 (a_{43} c_2 + a_{43} c_3) &= \frac{1}{8} \end{aligned} \quad (2.3)$$

$$b_3 a_{32} + b_4 a_{42} = b_2(1 - c_2)$$

$$b_4 a_{43} = b_3(1 - c_3)$$

$$b_4(1 - c_4) = 0$$

Las elecciones de Kutta que han llegado a ser particularmente populares son

$$I) \quad c_2 = c_3 = \frac{1}{2}, \quad b_1 = \frac{1}{6}, \quad b_3 = \frac{1}{3}, \quad b_2 = \frac{1}{3}, \quad b_4 = \frac{1}{6}$$

$$II) \quad c_2 = \frac{1}{3}, \quad c_3 = \frac{2}{3}$$

Ambos métodos generalizan las reglas de cuadraturas clásicas obteniendo el mismo orden. El primero es más popular, el segundo más preciso. Están dados como se indica enseguida

$$\begin{array}{c|cccc} 0 & & & & \\ \hline \frac{1}{2} & \frac{1}{2} & & & \\ \frac{1}{2} & 0 & \frac{1}{2} & & \\ 1 & 0 & 0 & 1 & \\ \hline \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6} & \end{array}$$

$$\begin{array}{c|cccc} 0 & & & & \\ \hline \frac{1}{3} & \frac{1}{3} & & & \\ \frac{2}{3} & -\frac{1}{3} & 1 & & \\ 1 & 1 & -1 & 1 & \\ \hline \frac{1}{6} & \frac{2}{6} & \frac{2}{6} & \frac{1}{6} & \end{array}$$

$$k_1 = f(x_0, y_0)$$

$$k_2 = f(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}hk_1)$$

$$k_3 = f(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}hk_2)$$

$$k_4 = f(x_0 + h, y_0 + hk_3)$$

$$y_1 = y_0 + h(\frac{1}{6}k_1 + \frac{2}{6}k_2 + \frac{2}{6}k_3 + \frac{1}{6}k_4)$$

$$k_1 = f(x_0, y_0)$$

$$k_2 = f(x_0 + \frac{1}{3}h, y_0 + \frac{1}{3}hk_1)$$

$$k_3 = f(x_0 + \frac{2}{3}h, y_0 + h(-\frac{1}{3}k_1 + k_2))$$

$$k_4 = f(x_0 + h, y_0 + h(k_1 - k_2 + k_3))$$

$$y_1 = y_0 + h(\frac{1}{8}k_1 + \frac{3}{8}k_2 + \frac{3}{8}k_3 + \frac{1}{8}k_4)$$

En un método Runge-Kutta nos interesa proporcionar cotas para el error local [8], a saber:

Teorema

Si un método de Runge-Kutta es de orden p y si todas las derivadas parciales de $f(x, y)$ hasta de orden p existen y son continuas entonces

$$\|y(x_0 + h) - y_1\| \leq h^{p+1} \left[\frac{1}{(p+1)!} \max_{t \in [0,1]} \|y^{p+1}(x_0 + th)\| + \frac{1}{p!} \sum_{i=1}^p |b_i| \max_{t \in [0,1]} \|k_i^p(th)\| \right]$$

es decir

$$\|y(x_0 + h) - y_1\| \leq Ch^{p+1}$$

En el caso vectorial, la relación anterior es válida componente a componente

$$|y^j(x_0 + h) - y_1^j| = Ch^{p+1} + O(h^{p+2}) \quad j = 1, 2, \dots, n$$

donde C depende de los coeficientes del método.

Estas estimaciones del error son de poco interés práctico debido a que requieren el cálculo y obtención del máximo de varias derivadas de ordenes altos de $f(x, y)$. Pero la ventaja principal de los métodos Runge-Kutta respecto a las series de Taylor es precisamente que el cálculo de las derivadas no son necesarias. Sin embargo, de cualquier forma requerimos estimaciones del error; primero para asegurar que los tamaños de paso h_i sean escogidos lo suficientemente pequeños para asegurar la precisión requerida, y segundo para garantizar que h_i es lo suficientemente grande para evitar trabajo computacional innecesario.

Existen muchas ideas para lograr lo anterior. La más vieja quizá, es debida a Runge que consiste en repetir el cálculo con el tamaño de paso reducido a la mitad y luego comparar ambos resultados: los dígitos que no cambian se toman como verdaderos. La siguiente idea debida a Richardson, hace uso del comportamiento del error como función de h .

Dada la condición inicial (x_0, y_0) y tamaño de paso h evaluemos dos veces usando un método de Runge-Kutta de orden p obteniendo los resultados numéricos y_1 y y_2 . Hagamos nuevamente otro cálculo con la misma condición inicial pero con paso $2h$ otorgándonos la solución w .

Tenemos que el error de y_1 es

$$e_1 = y(x_0 + h) - y_1 = Ch^{p+1} + O(h^{p+2})$$

El error cometido en y_2 consiste de dos partes: el que se acarrea de y_1 y el error local del segundo paso. Si se toma en cuenta lo anterior se obtiene [6]:

$$e_2 = 2Ch^{p+1} + O(h^{p+2})$$

Cuando tomamos el paso $2h$ el error es

$$y(x_0 + 2h) - w = C(2h)^{p+1} + O(h^{p+2})$$

Ignorando los términos $O(h^{p+2})$, eliminamos el parámetro C de las dos últimas ecuaciones obteniendo una mejor aproximación \hat{y}_2 para $y(x_0 + 2h)$. Dicho más formalmente

Teorema

Supóngase que y_2 es el resultado numérico de dos evaluaciones con paso h por medio de un método tipo Runge-Kutta de orden p y w el resultado por el mismo método pero con paso $2h$. Entonces el error de y_2 puede ser extrapolado de la siguiente forma

$$y(x_0 + 2h) - y_2 = \frac{y_2 - w}{2^p - 1} + O(h^{p+1}) \quad (2.4)$$

y

$$\hat{y}_2 = y_2 + \frac{y_2 - w}{2^p - 1}$$

es una aproximación de orden $p + 1$ a $y(x_0 + 2h)$.

Las fórmulas anteriores son muy fáciles de calcular. La primera nos da una estimación del error cometido y la segunda incrementa el orden en uno.

2.1. Control automático del paso

Una estimación del error de (2.4) lo podemos obtener como

$$err = \frac{1}{2^p - 1} \max_{i=1, \dots, n} \frac{|y_2^i - w_i|}{d_i}$$

Con frecuencia son usadas otras normas, como la euclídeana por ejemplo. La cantidad d_i es un factor de escalamiento: $d_i = 1$ para errores absolutos y $d_i = |y_1^i|$ para errores relativos.

El nuevo paso h_{nueva} será aceptado ó rechazado dependiendo de si el error cometido es menor ó no que una tolerancia tol dada. Si el paso no es aceptado debe ser reducido, pero por motivos de eficiencia buscamos que sea lo más grande posible. Luego, $h_{nueva} = \theta h$ debe ser tal que el error cometido se comporte como el anterior y que sea menor que tol . Es decir,

$$C(\theta h)^{p+1} \leq tol = \left(\frac{tol}{err}\right) err$$

$$(Ch^{p+1}) \theta^{p+1} \leq \left(\frac{tol}{err}\right) err$$

de donde, escogiendo

$$\theta^{p+1} \leq \frac{tol}{err} \quad (\text{i.e.}) \quad \theta \leq \left(\frac{tol}{err}\right)^{\frac{1}{p+1}}$$

Entonces, si $err \leq tol$ la solución avanza a y_2 o bien al punto \hat{y}_2 y se intenta avanzar tomando como h_{nueva} el paso actual. En caso contrario el cálculo se repite con h_{nueva} .

El paso máximo h_{max} generalmente se escoge entre 1.5 y 5 con el objeto de prevenir que el paso actual llegue a tomar valores muy grandes. Si en el paso actual $err \geq tol$ se aconseja tomar $h_{max} = 1$.

También es claro que para pasos muy pequeños el trabajo computacional se incrementa enormemente, por lo que se tiene que seleccionar un tamaño mínimo h_{min} .

En general, el tamaño del paso h_{nueva} está dado por una función del tipo

$$h_{nueva} = h \cdot \theta$$

$$\theta = \min \left(facmax, \max \left(facmin, fac \left(\frac{tol}{err} \right)^{\frac{1}{p+1}} \right) \right)$$

donde $facmax, facmin, fac$ son números que se eligen en forma experimental para aumentar la eficiencia del método. Para una discusión amplia sobre este punto véase Shampine-Watts[11]

Una tercera idea es construir fórmulas de Runge-Kutta que contengan, además de la aproximación numérica y_1 , una expresión \hat{y}_1 de orden mayor. Esto puede servir para estimar el error y controlar el tamaño del paso.

Luego, nuestro propósito es encontrar un esquema de coeficientes

0					
c_2	a_{21}				
c_3	a_{31}	a_{32}			
\vdots	\vdots	\vdots			
c_r	a_{r1}	a_{r2}	\dots	$a_{r(r-1)}$	
	b_1	b_2		b_{r-1}	b_r
	b_1	b_2		b_{r-1}	b_r

tal que

$y_1 = y_0 + h(b_1 k_1 + \dots + b_s k_s)$ sea de orden p y

$\hat{y}_1 = y_0 + h(\hat{b}_1 k_1 + \dots + \hat{b}_s k_s)$ sea de orden q , por lo general $q = p + 1$

El primer método de este tipo fue propuesto por Merson (1957), Ceshino (1962) y Zonneveld (1963). Entenderemos "p(q)" como que y_1 es de orden p y \hat{y}_1 de orden q

0				
$\frac{1}{4}$	$\frac{1}{4}$			
$\frac{1}{2}$	0	$\frac{1}{2}$		
1	1	-2	2	
	1	-2	2	
	$\frac{1}{6}$	0	$\frac{4}{6}$	$\frac{1}{6}$

0					
$\frac{1}{3}$	$\frac{1}{3}$				
$\frac{1}{2}$	$\frac{1}{6}$	$\frac{1}{6}$			
$\frac{1}{2}$	$\frac{1}{6}$	0	$\frac{3}{6}$		
1	$\frac{1}{2}$	0	$-\frac{3}{2}$	2	
	$\frac{1}{2}$	0	$-\frac{3}{2}$	2	
	$\frac{1}{6}$	0	$\frac{2}{6}$	$\frac{1}{6}$	

$$k_1 = f(x_0, y_0)$$

$$k_2 = f\left(x_0 + \frac{1}{4}h, y_0 + \frac{1}{4}hk_1\right)$$

$$k_3 = f\left(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}hk_2\right)$$

$$k_4 = f\left(x_0 + h, y_0 + h(k_1 - 2k_2 + 2k_3)\right)$$

$$y_1 = y_0 + h(k_1 - 2k_2 + 2k_3)$$

$$\hat{y}_1 = y_0 + h\left(\frac{1}{6}k_1 + \frac{4}{6}k_3 + \frac{1}{6}k_4\right)$$

$$k_1 = f(x_0, y_0)$$

$$k_2 = f\left(x_0 + \frac{1}{3}h, y_0 + \frac{1}{3}hk_1\right)$$

$$k_3 = f\left(x_0 + \frac{1}{3}h, y_0 + h\left(\frac{1}{6}k_1 + \frac{1}{6}k_2\right)\right)$$

$$k_4 = f\left(x_0 + \frac{1}{2}h, y_0 + h\left(\frac{1}{8}k_1 + \frac{3}{8}k_2\right)\right)$$

$$k_5 = f\left(x_0 + h, y_0 + h\left(\frac{1}{2}k_1 - \frac{3}{2}k_2 + 2k_4\right)\right)$$

$$y_1 = y_0 + h\left(\frac{k_1}{2} - \frac{3}{2}k_2 + 2k_4\right)$$

$$\hat{y}_1 = y_0 + h\left(\frac{1}{6}k_1 + \frac{2}{3}k_4 + \frac{1}{6}k_5\right)$$

Ceshino 2(4)

Merson 4(5)

ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA

0				
$\frac{1}{2}$	$\frac{1}{2}$			
$\frac{1}{2}$	0	$\frac{1}{2}$		
1	0	0	1	
$\frac{3}{4}$	$\frac{6}{32}$	$\frac{7}{32}$	$\frac{13}{32}$	$-\frac{1}{32}$
	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{6}$
	$-\frac{1}{2}$	$\frac{7}{3}$	$\frac{7}{3}$	$-\frac{16}{3}$

$$k_1 = f(x_0, y_0)$$

$$k_2 = f\left(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}hk_1\right)$$

$$k_3 = f\left(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}hk_2\right)$$

$$k_4 = f(x_0 + h, y_0 + hk_3)$$

$$k_5 = f\left(x_0 + \frac{3}{4}h, y_0 + h\left(\frac{5}{32}k_1 + \frac{7}{32}k_2 + \frac{13}{32}k_3 - \frac{1}{36}k_4\right)\right)$$

$$y_1 = y_0 + h\left(\frac{1}{6}k_1 + \frac{1}{3}k_2 + \frac{1}{3}k_3 + \frac{1}{6}k_4\right)$$

$$\hat{y}_1 = y_0 + h\left(-\frac{1}{2}k_1 + \frac{7}{3}k_2 + \frac{7}{3}k_3 - \frac{16}{3}k_5\right)$$

Zonneveld 4(3)

Sin embargo, ninguno de estos métodos satisfacen nuestras condiciones. En el método de Merson la aproximación \hat{y}_1 es de orden 5 solamente para ecuaciones lineales, sobrestimando el error para h pequeña. De manera similar el método de Zonneveld no calcula el error de truncamiento y el de Ceshino no es nada económico debido a que realiza un cálculo muy preciso del error.

Métodos que satisfacen nuestras condiciones han sido dados por Sarafyan (1966), England (1969) y Fehlberg (1963, 1969). Haremos la derivación de las fórmulas de England que en particular es el método que elegimos para resolver nuestro problema.

En el método de England una fórmula de cuarto orden es usada para avanzar la solución dos pasos. Una evaluación adicional de la derivada es hecha antes de completar el segundo paso que nos permite estimar el error local acumulado en ambos

pesos. El procedimiento realiza 9 evaluaciones y si el peso es rechazado 7 de ellas serán deshechadas. Más precisamente:

$$\begin{aligned}k_0 &= hf(x_0, y_0) \\k_1 &= hf\left(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}k_0\right) \\k_2 &= hf\left(x_0 + \frac{1}{2}h, y_0 + \frac{1}{4}(k_0 + k_1)\right) \\k_3 &= hf(x_0 + h, y_0 - k_1 + 2k_2) \\y_1 &= y_0 + \frac{1}{8}(k_0 + 4k_2 + k_3)\end{aligned}$$

y se calcula el segundo paso en la segunda forma

$$\begin{aligned}k_4 &= hf(x_0 + h, y_1) \\k_5 &= hf\left(x_0 + \frac{3}{2}h, y_1 + \frac{1}{2}k_4\right) \\k_6 &= hf\left(x_0 + \frac{3}{2}h, y_1 + \frac{1}{4}(k_4 + k_5)\right)\end{aligned}$$

antes de terminar el segundo paso se realiza una evaluación extra para poder calcular el error local en los dos pasos

$$\begin{aligned}k_7 &= hf\left(x_0 + 2h, y_0 + \frac{1}{6}(-k_0 - 96k_1 + 92k_2 - 121k_3 + 144k_4 + 6k_5 - 12k_6)\right) \\err &= \frac{1}{90}(-k_0 + 4k_2 + 17k_3 - 23k_4 + 4k_5 - k_7)\end{aligned}$$

Si $err \leq tol$ entonces terminamos el cálculo del segundo paso

$$\begin{aligned}k_8 &= hf(x_0 + 2h, y_1 - k_5 + 2k_6) \\y_2 &= y_1 + \frac{1}{4}(k_4 + 4k_6 + k_8)\end{aligned}$$

Si el tamaño de paso h es aceptado, un total de 4¹ evaluaciones por paso son hechas, en caso contrario tenemos que deshechar 1. Lo que hace al esquema de England mucho más flexible que el del paso doble ya que podemos reducir h por un factor adecuado sin tener que pagar un costo alto. El esquema de England [12] es el siguiente

	0								
	$\frac{1}{2}$	$\frac{1}{2}$							
	$\frac{1}{2}$	$\frac{1}{6}$	$\frac{1}{4}$						
	1	0	-1	2					
Solución de cuarto orden en $x+h$		$\frac{1}{6}$	0	$\frac{2}{3}$	$\frac{1}{6}$				
	1	$\frac{1}{6}$	0	$\frac{2}{3}$	$\frac{1}{6}$				
	$\frac{2}{3}$	$\frac{1}{6}$	0	$\frac{2}{3}$	$\frac{1}{6}$	$\frac{1}{2}$			
	$\frac{3}{2}$	$\frac{1}{6}$	0	$\frac{2}{3}$	$\frac{1}{6}$	$\frac{1}{4}$	$\frac{1}{6}$		
	2	$\frac{1}{6}$	0	$\frac{2}{3}$	$\frac{1}{6}$	0	-1	2	
	2	$-\frac{1}{6}$	-16	$\frac{46}{3}$	$-\frac{111}{6}$	24	1	-2	0
Solución de cuarto orden en $x+2h$		$\frac{1}{6}$	0	$\frac{2}{3}$	$\frac{1}{6}$	$\frac{1}{6}$	0	$\frac{2}{3}$	0
	err	$\frac{1}{90}$	0	$-\frac{4}{90}$	$-\frac{17}{90}$	$\frac{23}{90}$	0	$-\frac{4}{90}$	$\frac{1}{90}$

$$k_0 = f(x_0, y_0)$$

$$k_1 = f\left(x_0 + \frac{1}{2}h, y_0 + \frac{1}{2}hk_0\right)$$

$$k_2 = f\left(x_0 + \frac{1}{2}h, y_0 + \frac{1}{4}h(k_0 + k_1)\right)$$

$$k_3 = f\left(x_0 + h, y_0 + h(-k_1 + 2k_2)\right)$$

$$y_1 = y_0 + \frac{1}{6}h(k_0 + 4k_2 + k_3)$$

$$k_4 = f(x_0 + h, y_1)$$

$$k_5 = f\left(x_0 + \frac{3}{2}h, y_1 + \frac{1}{2}hk_4\right)$$

$$k_6 = f\left(x_0 + \frac{3}{2}h, y_1 + \frac{1}{4}h(k_4 + k_5)\right)$$

$$k_7 = f\left(x_0 + 2h, y_0 + \frac{1}{6}h(-k_0 - 96k_1 + 92k_2 - 121k_3 + 144k_4 + 6k_5 - 12k_6)\right)$$

$$err = \frac{1}{90}h(-k_0 + 4k_2 + 17k_3 - 23k_4 + 4k_5 - k_7)$$

$$k_8 = f(x_0 + 2h, y_1 + h(-k_5 + 2k_6))$$

$$y_2 = y_1 + \frac{1}{6}h(k_4 + 4k_6 + k_8)$$

CAPÍTULO IV

CÁLCULO DE CICLOS LÍMITE

Supóngase que el sistema autónomo

$$(I) \quad \begin{aligned} \frac{dx}{dt} &= f_1(x, y) \\ \frac{dy}{dt} &= f_2(x, y) \end{aligned}$$

tiene un ciclo límite de período P . Es decir, el sistema posee una solución periódica que es un ciclo límite.

Uno de los objetivos fundamentales de este trabajo es proporcionar un método interactivo para determinar el período de un ciclo límite.

Ejemplo

Sea

$$\begin{aligned} x' &= y \\ y' &= \alpha x(1 - x^2) - x \end{aligned}$$

este sistema describe el oscilador de Van der Pol y se sabe que para $\alpha > 0$ tiene un ciclo límite y también que una aproximación al período en función de α está dado por

$$P(\alpha) = 2\pi \left(1 + \frac{\alpha^2}{8} + \dots \right)$$

En general no es fácil determinar analíticamente el período para el sistema (I), sin embargo, veremos que numéricamente puede ser determinado de manera económica.

1. Planteamiento del Problema

Determinar numéricamente un punto (x_0, y_0) perteneciente al ciclo límite de un sistema autónomo de ecuaciones diferenciales en el plano y su período.

Un primer intento de abordar el problema consistió en calcular un punto fijo del mapeo de Poincaré.

Si (x_0, y_0) es un punto sobre el ciclo límite de período P , entonces para cada (x, y) perteneciente a una sección transversal Σ a la solución en (x_0, y_0) , existe un $r(x, y)$ tal que la solución $\phi(t; x, y, P)$ corta a Σ en $t = r(x, y)$ si (x, y) pertenece a una vecindad de (x_0, y_0)

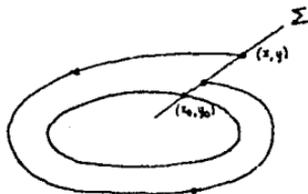


Fig. 1

Luego, nos interesa determinar (x_0, y_0) tal que

$$\phi(r(x_0, y_0); x_0, y_0, P) - (x_0, y_0) = 0$$

con $r(x_0, y_0) = P > 0$

Reescribiendo este sistema en la forma

$$g_1(x_0, y_0, P) = \phi_1(r(x_0, y_0); x_0, y_0, P) - x_0 = 0$$

$$g_2(x_0, y_0, P) = \phi_2(r(x_0, y_0); x_0, y_0, P) - y_0 = 0$$

donde $r(x_0, y_0)$ es el tiempo que tarda la solución que empieza en $(x_0, y_0) \in \Sigma$ en regresar a Σ .

Tomando $\Sigma = \{(x_0, y_0) : y_0 = c\}$ entonces el sistema queda

$$g_1(x_0, c, P) = \phi_1(r(x_0, c); x_0, c, P) - x_0 = 0$$

$$g_2(x_0, c, P) = \phi_2(r(x_0, c); x_0, c, P) - c = 0$$



Fig. 2

de donde

$$\frac{\partial}{\partial x_0} g_1(x_0, c, P) = \frac{\partial}{\partial x_0} \phi_1(r(x_0, c); x_0, c, P) + \frac{\partial}{\partial r} \phi_1((r(x_0, c); x_0, c, P) \frac{\partial}{\partial x_0} r(x_0, c) - 1 = 0$$

$$\frac{\partial}{\partial x_0} g_2(x_0, c, P) = \frac{\partial}{\partial x_0} \phi_2(r(x_0, c); x_0, c, P) + \frac{\partial}{\partial r} \phi_2((r(x_0, c); x_0, c, P) \frac{\partial}{\partial x_0} r(x_0, c) = 0$$

De esta última ecuación obtenemos¹

$$\frac{\partial r}{\partial x_0} = - \frac{\frac{\partial \phi_2}{\partial x_0}}{\frac{\partial \phi_2}{\partial r}} \quad (1.1)$$

Luego, para encontrar x_0 aplicamos el método de Newton a $g_1(x_0, c, P)$, por lo que nos hace falta calcular $\frac{\partial \phi_1}{\partial x_0}$ y $\frac{\partial \phi_2}{\partial x_0}$ que pueden ser obtenidas de la siguiente manera:

$$\text{Como } \frac{d\phi_1}{dt} = f_1(\phi_1, \phi_2) \text{ y } \frac{d\phi_2}{dt} = f_2(\phi_1, \phi_2)$$

obtenemos

$$\begin{aligned} \frac{\partial}{\partial x_0} \left(\frac{d\phi_1}{dt} \right) &= \frac{\partial f_1}{\partial \phi_1} \frac{\partial \phi_1}{\partial x_0} + \frac{\partial f_1}{\partial \phi_2} \frac{\partial \phi_2}{\partial x_0} \\ \frac{\partial}{\partial x_0} \left(\frac{d\phi_2}{dt} \right) &= \frac{\partial f_2}{\partial \phi_1} \frac{\partial \phi_1}{\partial x_0} + \frac{\partial f_2}{\partial \phi_2} \frac{\partial \phi_2}{\partial x_0} \end{aligned}$$

¹ Se sobreentiende que las evaluaciones son en la solución

Es decir, $\frac{\partial \phi_1}{\partial x_0}$ y $\frac{\partial \phi_2}{\partial x_0}$ son soluciones de la Ecuación Variacional

$$\frac{d}{dt} \begin{pmatrix} \frac{\partial \phi_1}{\partial x_0} \\ \frac{\partial \phi_2}{\partial x_0} \end{pmatrix} = \begin{pmatrix} \frac{\partial f_1}{\partial \phi_1} & \frac{\partial f_1}{\partial \phi_2} \\ \frac{\partial f_2}{\partial \phi_1} & \frac{\partial f_2}{\partial \phi_2} \end{pmatrix} \begin{pmatrix} \frac{\partial \phi_1}{\partial x_0} \\ \frac{\partial \phi_2}{\partial x_0} \end{pmatrix}$$

con las condiciones iniciales

$$\begin{aligned} \phi_1(0, x_0, c, P) &= x_0 \\ \frac{\partial \phi_1}{\partial x_0}(0, x_0, c, P) &= 1 \\ \phi_2(0, x_0, c, P) &= c \\ \frac{\partial \phi_2}{\partial x_0}(0, x_0, c, P) &= 0 \end{aligned}$$

Entonces, se puede aplicar el método de Newton para encontrar x_0 a

$$g_1(x_0, c, P) = \phi_1(r(x_0, c); x_0, c, P) - x_0 = 0$$

y $r(x_0, c)$ lo obtenemos de (1.1).

Realmente se está resolviendo un sistema de ecuaciones en r y x_0 , pero sólo se trabaja explícitamente con x_0 , ya que es necesario determinar la $r(x_0, c)$ tal que $\phi_2(r(x_0, c); x_0, c, P) = c$ y esto va a depender del método que se use para integrar las ecuaciones diferenciales. Este método se reporta en la literatura por Curry [8] para encontrar soluciones periódicas para la ecuación de Lorentz, pero no indica el método de integración usado. En nuestro caso usamos el método Runge-Kutta-England (RKE) [1] que no nos permitió determinar $r(x_n, c)$ para ningún n .

Lo anterior aunado a la necesidad de resolver las ecuaciones variacionales y lo difícil que resultó trabajarlo interactivamente decidimos hacer uso de otra idea [9].

Transformemos el sistema (I) en otro en el cual el ciclo límite tenga período 1 ([9]). Para ello hagamos el cambio de variable $t = Px$ donde P es el período que queremos determinar. Sustituyendo en (I) obtenemos

$$(II) \quad \begin{aligned} \frac{dx}{dz} &= Pf_1(x, y) \\ \frac{dy}{dz} &= Pf_2(x, y) \end{aligned}$$

con $z \in [0, 1]$

Usualmente se tienen dos formas de abordar el problema (II) desde el punto de vista numérico.

2. Aproximación por diferencias.

Si el sistema (II) lo escribimos en la forma

$$(II') \quad \frac{dX}{dz} = PF(X)$$

Aproximando por diferencias $\frac{dX}{dz}$ en los puntos

$$z_0 = 0, z_1 = h, \dots, z_N = Nh = 1$$

obtenemos algo de la forma

$$\frac{X^{i+1} - X^i}{h} = PF\left(\frac{1}{2}(X^{i+1} + X^i)\right) \quad i = 0, 1, \dots, N-1$$

donde $X^i = X(z_i)$

Como queremos una solución periódica tenemos la condición adicional

$$X^0 = X^N$$

Luego, tenemos un sistema de $2(N+1)$ ecuaciones con $2(N+1)+1$ incógnitas

$$X^0, X^1, \dots, X^N, P$$

por lo que es necesario fijar una variable que no sea P , por supuesto, y resolver el sistema correspondiente por algún método como Newton o cuasi-Newton.

3. Métodos de tiro.

Estos métodos están basados en métodos que calculan numéricamente las soluciones de la ecuación diferencial, dadas las condiciones iniciales. La idea es la siguiente:

Elegimos un punto inicial (x_0, y_0) y proponemos un valor para P , entonces el sistema (II) puede integrarse de $z = 0$ a $z = 1$, o bien, de $z = 0$ a $z = -1$ dependiendo de la estabilidad del ciclo límite. Supondremos que la integración se realizó de $z = 0$ a $z = 1$, luego en $z = 1$ obtenemos los valores

$$x(1) = \phi_1(1; x_0, y_0, P)$$

$$x(2) = \phi_2(1; x_0, y_0, P)$$

que dependen de la elección de x_0, y_0 y P . En el caso de una solución periódica debemos tener que satisfacen las ecuaciones

$$g_1(x_0, y_0, P) = 0$$

$$g_2(x_0, y_0, P) = 0$$

donde

$$g_1(x_0, y_0, P) = \phi_1(r(x_0, y_0); x_0, y_0, P) - x_0$$

$$g_2(x_0, y_0, P) = \phi_2(r(x_0, y_0); x_0, y_0, P) - y_0$$

Luego, tenemos dos ecuaciones con tres incógnitas, por consiguiente tenemos que fijar una de ellas, la cuál evidentemente no puede ser P . Lo que es claro geométricamente, ya que si x_0 se mantiene fija es necesario que la recta $x = x_0$ corte a la solución que andamos buscando (figura 3) de otra manera el problema tendría como solución $P = 0$.

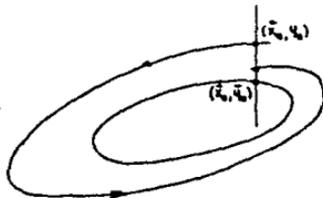


Fig. 3

Así que la elección del punto inicial (x_0, y_0) es importante para que tenga éxito el método. El sistema se puede resolver usando el método de Newton o un cuasi-Newton.

En este trabajo estamos interesados en los métodos de tiro, ya que contamos con un método para calcular las soluciones de un sistema de ecuaciones diferenciales con condiciones iniciales y también en que disponemos fácilmente del espacio fase. Además el sistema de ecuaciones es un sistema de 2×2 comparado al método de aproximación por diferencias que puede darnos un sistema de muchas variables, pues depende del número de puntos N sobre la trayectoria que puede ser muy grande para obtener una buena aproximación.

Es usual que en el método de tiro se utilice el método de Newton para resolver el sistema de ecuaciones no-lineales resultante, sin embargo, esto hace que el costo total sea excesivo debido a la necesidad de resolver las ecuaciones variacionales para poder

obtener una aproximación al Jacobiano, por lo que nosotros preferimos usar el método de Broyden.

El sistema a resolver está dado por

$$\begin{aligned}g_1(x_0, y, P) &= \phi_1(1; x_0, y, P) - x_0 = 0 \\g_2(x_0, c, P) &= \phi_2(1; x_0, y, P) - y = 0\end{aligned}$$

en forma más compacta

$$G(u) = \begin{pmatrix} g_1(u) \\ g_2(u) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

donde $u = (y, P)^T$

Inicialmente se eligen una aproximación (x_0, y_0) de un punto sobre el ciclo límite y una estimación del período P_0 , con esto se puede calcular la solución del sistema:

$$\begin{aligned}x' &= P_0 f_1(x, y) \\y' &= P_0 f_2(x, y)\end{aligned}$$

con condición inicial $(x_0, y_0)^T$

Si llamamos a la solución

$$\{\phi_1(t; x_0, y_0, P_0), \phi_2(t; x_0, y_0, P_0)\}$$

de aquí se obtiene el valor de la función

$$G(u_0) = \begin{pmatrix} \phi_1(1; x_0, y_0, P_0) - x_0 \\ \phi_2(1; x_0, y_0, P_0) - y_0 \end{pmatrix}$$

con $u_0 = (y_0, P_0)^T$

La aproximación inicial J_0 al Jacobiano en u_0 la obtenemos de la siguiente manera

Si $J_0 = (a_1 | a_2)$ con

$$a_1 = \frac{G(u_0 + \epsilon e_1) - G(u_0)}{\epsilon}$$

$$a_2 = \frac{G(u_0 + \epsilon e_2) - G(u_0)}{\epsilon}$$

donde e_1, e_2 son los vectores canónicos y $\epsilon =$ raíz cuadrada de la precisión del sistema de punto flotante empleado. Es decir, perturbamos y_0 y calculamos la solución del sistema

$$\begin{aligned}x' &= P_0 f_1(x, y) \\ y' &= P_0 f_2(x, y)\end{aligned}$$

con condición inicial $(x_0, y_0 + \epsilon)^T$. Luego, calculamos

$$G(u_0 + \epsilon e_1) = \begin{pmatrix} \phi_1(1; x_0, y_0 + \epsilon, P_0) - x_0 \\ \phi_2(1; x_0, y_0 + \epsilon, P_0) - y_0 + \epsilon \end{pmatrix}$$

Para obtener $G(u_0 + \epsilon e_2)$ se calcula la solución del sistema

$$\begin{aligned}x' &= (P_0 + \epsilon) f_1(x, y) \\ y' &= (P_0 + \epsilon) f_2(x, y)\end{aligned}$$

con condición inicial (x_0, y_0) y entonces

$$G(u_0 + \epsilon e_2) = \begin{pmatrix} \phi_1(1; x_0, y_0, P_0 + \epsilon) - x_0 \\ \phi_2(1; x_0, y_0, P_0 + \epsilon) - y_0 \end{pmatrix}$$

Solamente nos resta indicar que no necesitamos resolver el sistema de ecuaciones para calcular la dirección de descenso p_k , debido a que en el método de Broyden

$$A_{k+1} = A_k + u_k v_k^T$$

con

$$\begin{aligned}u_k &= y_k - A_k p_k \\ v_k^T &= \frac{p_k^T}{p_k^T p_k}\end{aligned}$$

ya que si conocemos la inversa de A_k es fácil obtener A_{k+1}^{-1} como lo indica la fórmula de Sherman-Morrison-Woodbury:

Lema

Sean $u, v \in \mathbb{R}^n$, $A \in M_{n \times n}(\mathbb{R})$ no-singular. Entonces

$$A + uv^T \text{ es no-singular} \iff 1 + v^T A^{-1} u = \sigma \neq 0$$

Además

$$(A + uv^T)^{-1} = A^{-1} - \frac{1}{\sigma} A^{-1} uv^T A^{-1}$$

Demostración:

\Rightarrow)

Supóngase que $A + uv^T$ es no-singular. Como A es no-singular, entonces $u^T (A^{-1})^T u > 0$ para todo $u \in \mathbb{R}^n$, $u \neq 0$. Se tiene que $x^T (A + uv^T) x > 0$ para $x \neq 0$, en particular para $x = A^{-1}u$. Por tanto

$$\begin{aligned} (A^{-1}u)^T (A + uv^T) (A^{-1}u) &= (A^{-1}u)^T (u + uv^T A^{-1}u) \\ &= (A^{-1}u)^T u (1 + v^T A^{-1}u) > 0 \end{aligned}$$

De donde

$$1 + v^T A^{-1}u \neq 0$$

\Leftarrow

Supongamos ahora que $1 + v^T A^{-1}u \neq 0$. Tenemos que

$$\begin{aligned} (A + uv^T) \left(A^{-1} - \frac{1}{\sigma} A^{-1} uv^T A^{-1} \right) &= \\ I - \frac{1}{\sigma} uv^T A^{-1} - \frac{1}{\sigma} uv^T A^{-1} uv^T A^{-1} + uv^T A^{-1} &= \\ I + uv^T A^{-1} \left(I - \frac{1}{\sigma} I - \frac{1}{\sigma} v^T A^{-1} u \right) &= \\ I + uv^T A^{-1} \left(I - \frac{1 + v^T A^{-1} u}{\sigma} I \right) &= I \end{aligned}$$

□

Dado lo anterior, sólo es necesario calcular la inversa de la matriz original A_0 para obtener A_k^{-1} para todo k

Ahora ya es posible aplicar el método de Broyden como se explicó en el capítulo II.

El criterio de parada que usamos es

$$\|x_{k+1} - x_k\| \leq \|x_k\| 10^{-3}$$

Ejemplo

En el sistema de ecuaciones

$$x' = y^2 - (x+1) \left((x-1)^2 + \frac{1}{2} \right)$$

$$y' = -xy$$

localizamos un ciclo límite con condición inicial

$$P_0 = 4.484$$

$$x_0 = (-0.515, -1.433)$$

obteniendo el resultado

$$P = 4.505$$

$$x = (-0.515, -1.462)$$

CAPÍTULO V

UN SISTEMA PARA CALCULAR CICLOS LÍMITE EN EL PLANO

1. Definición

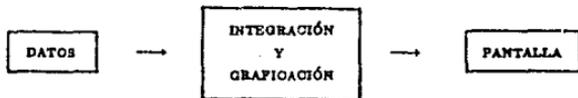
El sistema R.F es un programa que calcula y grafica trayectorias del retrato fase de un sistema autónomo de ecuaciones diferenciales ordinarias de la forma

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

así mismo, también es posible dibujar el campo direccional, imprimir gráficas o guardarlas en disco y calcular ciclos límite en caso de que existan.

Es posible tratar sistemas no-autónomos con la restricción de que siempre se empieza a integrar a partir de $t = 0$. Para su uso se requiere una computadora PC o compatible con tarjeta de graficación CGA.

R.F está integrado por las unidades mostradas en la siguiente figura



En "Datos" es donde residen los datos; El sistema dispondrá de ellos con una precisión de a lo más de cinco cifras significativas. del sistema. El bloque "Integración y graficación" es el que resuelve el problema planteado. La entidad "Pantalla" representa la fase entre el usuario y el sistema, ya que a través de la pantalla el usuario activa el sistema, introduce o modifica datos y obtiene el resultado deseado.

2. Bloque "Integración y graficación"

Realiza las siguientes funciones principales:

- 1) Integra numéricamente el sistema de ecuaciones diferenciales

ii) Manda a pantalla las soluciones

3. Requerimientos

a) El sistema se activará cuando el usuario escriba en la pantalla
>RF

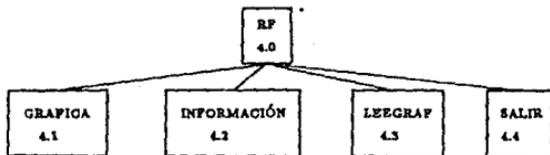
que corresponde a la iniciación del programa ejecutable RF.EXE

b) los datos de entrada al sistema serán otorgados interactivamente por el usuario que corresponden a:

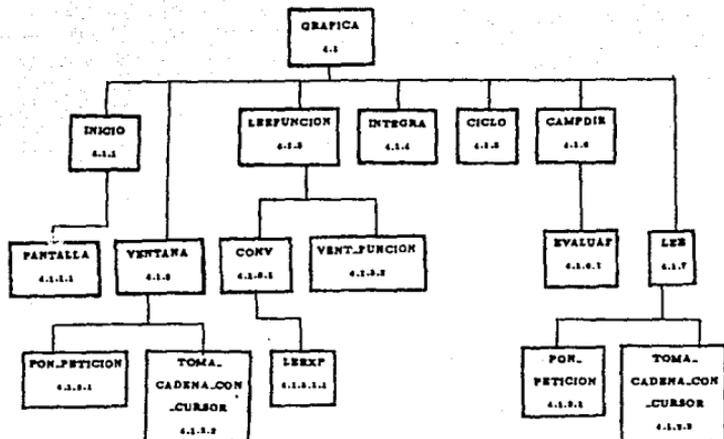
- El sistema de ecuaciones diferenciales a resolver.
- Región del plano donde se desee la solución.
- Condiciones iniciales.
- Nombres de archivos para guardar gráficas.

4. Diseño del sistema

RF.- Programa principal que coordina la solución del problema. Los módulos principales de este programa son cuatro, como se muestra enseguida



4.1 Grafica.- Este procedimiento otorga la solución del sistema de ecuaciones y la manda a la pantalla, para lo cual se encarga de leer interactivamente de la pantalla los datos de entrada proporcionados por el usuario. Hace uso del paquete de graficación Turbo-Graphix Versión 4. Los procedimientos que llama son los que se muestran en la siguiente figura

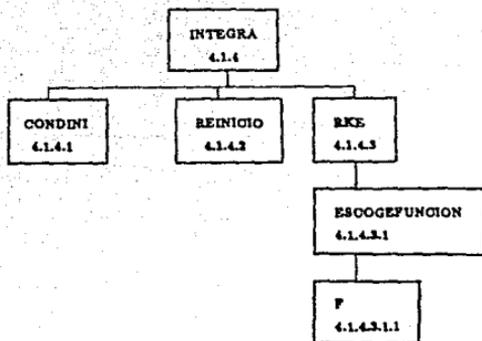


4.1.1 Inicio.- Contiene los datos iniciales. Para inicializar la pantalla se vale del procedimiento PANTALLA.

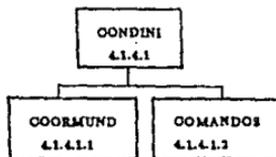
4.1.2 Ventana.- Solicita la región para graficar y el título de la gráfica. Usa los procedimientos PON_PETICION y TOMA_CADENA.CON_CURSOR para poner letreros y leer información en ambiente gráfico.

4.1.3 Leefuncion.- Procedimiento que solicita el sistema de ecuaciones por medio de LEEEXP en una región de la pantalla dada por VENT_FUNCION y verifica su sintaxis a través de CONV.

4.1.4 Integra.- Resuelve y grafica la ecuación diferencial. Integra hacia adelante ($t > 0$) ó con el tiempo en sentido contrario ($t < 0$) según sea solicitado. Utiliza los siguientes procedimientos:



4.1.4.1 Condini.- Lee las condiciones iniciales de la pantalla. Usa los procedimientos COORMUND y COMANDOS.



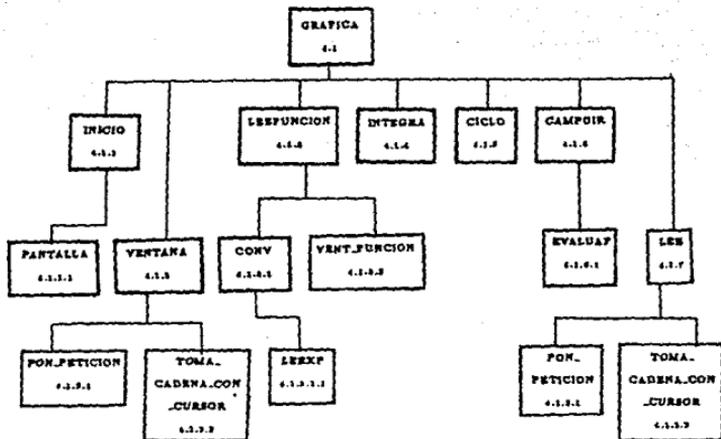
4.1.4.1.1 Coormund.- Realiza la transformación de coordenadas de pantalla a coordenadas reales.

4.1.4.1.2 Comandos.- Pone en pantalla todos los letreros para la interacción con el usuario.

4.1.4.2 Reinicio.- Reinicializa las condiciones iniciales según la tarea que se vaya a realizar.

4.1.4.3 Rke.- Integra el sistema de ecuaciones indicado por ESCOGEFUNCION. Emplea un método de tipo Runge-Kutta de órdenes 4 y 5 llamado Runge-Kutta-England.

4.1.4.3.1 Escogefuncion.- Elige el problema a resolver. El sistema de ecuaciones



4.1.1 Inicio.- Contiene los datos iniciales. Para inicializar la pantalla se vale del procedimiento PANTALLA.

4.1.2 Ventana.- Solicita la región para graficar y el título de la gráfica. Usa los procedimientos PON_PETICION y TOMA.CADENA.CON.CURSOR para poner letreros y leer información en ambiente gráfico.

4.1.3 Leefuncion.- Procedimiento que solicita el sistema de ecuaciones por medio de LEEXP en una región de la pantalla dada por VENT_FUNCION y verifica su sintaxis a través de CONV.

4.1.4 Integra.- Resuelve y grafica la ecuación diferencial. Integra hacia adelante ($t > 0$) ó con el tiempo en sentido contrario ($t < 0$) según sea solicitado. Utiliza los siguientes procedimientos:

otorgado por el usuario ó el modificado para calcular ciclos límite:

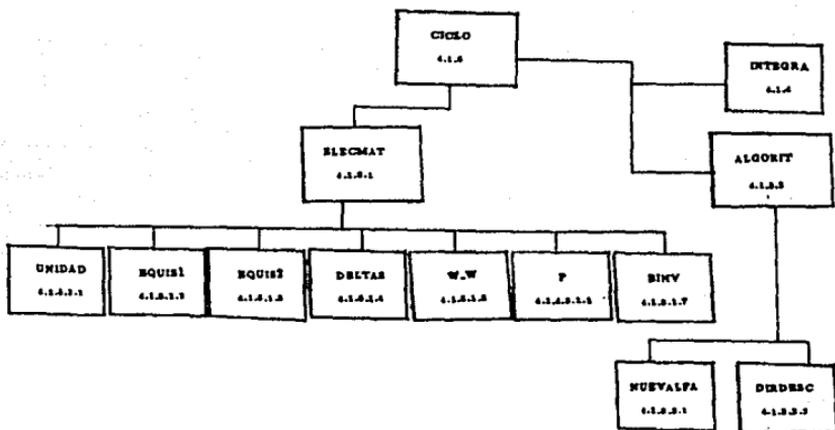
$$x' = Tf(x, y)$$

$$y' = Tg(x, y)$$

donde T es el período. Realiza la evaluación por medio del procedimiento F.

4.1.4.3.1.1 F.- Efectúa el cálculo del lado derecho del sistema de ecuaciones en el punto indicado.

4.1.5 Ciclo.- Calcula un punto del ciclo límite (en caso de que exista) y una aproximación de su período. Grafica cada una de las aproximaciones.



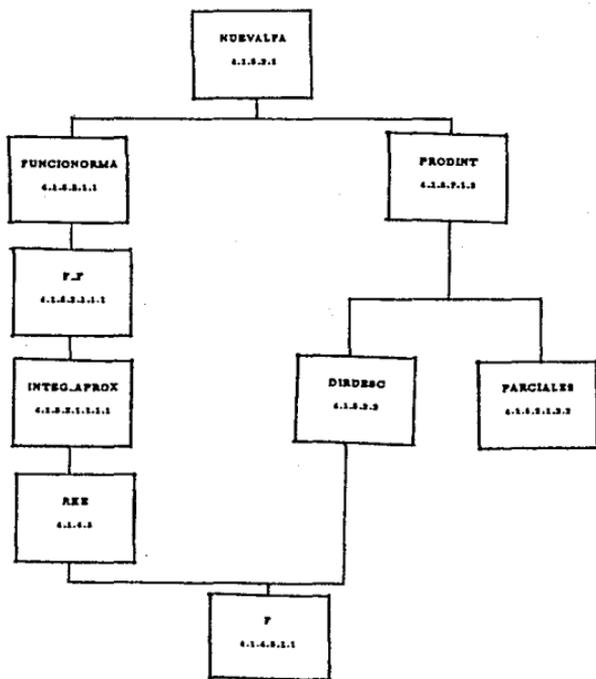
4.1.5.1 Eleccmat.- Elige el jacobiano inicial mediante una aproximación por diferencias finitas, para lo cual se vale de los procedimientos UNIDAD, EQUIS1, EQUIS2, DELTAS, W_W, F, BINV. UNIDAD calcula la precisión de la máquina. EQUIS1 y EQUIS2 son aproximaciones al primero y segundo puntos respectivamente. DELTAS proporciona las variaciones para calcular el jacobiano.

4.1.5.1.5 W_W.- Calcula la diferencia entre dos puntos consecutivos de la función $f(x) = \frac{1}{2} \|F(x)\|_2^2$

4.1.5.1.7 BInv.- Invierte la matriz inicial.

4.1.5.2 Algorit.- Otorga la nueva aproximación $x_{k+1} = x_k + \alpha_k p_k$.

4.1.5.2.1 NuevaAlfa.- Da el nuevo parámetro de búsqueda lineal α_k al aproximar la función f por un polinomio cúbico. Utiliza los procedimientos que se muestran en seguida.



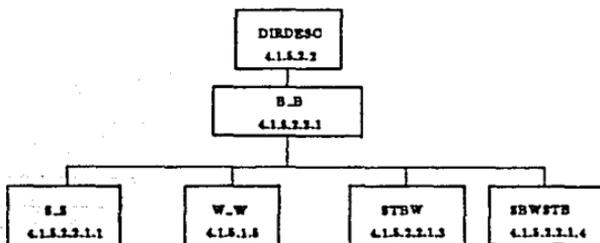
4.1.5.2.1.1 Funcionorma.- Calcula $f(x) = \frac{1}{2} \|F(x)\|_2^2$ por medio de F_F que a su vez usa INTEG.APROX.

4.1.5.2.1.1.1.1 Integ_aprox.- Integra el sistema de ecuaciones para el cálculo de ciclos límite en el intervalo $[0, 1]$.

4.1.5.2.1.2 Prodint.- Calcula $\nabla^T f(x_k) p_k$. Llama a los procedimientos F, DIRDESC

y PARCIALES.

4.1.5.2.2 DirDESC.- Calcula la dirección descendente p_1 . Usa los procedimientos B.B, S.S, W.W, STBW y SBWSTB.

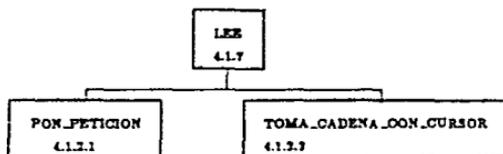


4.1.5.2.2.1 B.B.- Da la aproximación A_{k+1} del jacobiano.

4.1.5.2.2.2 Parciales.- Aproxima las parciales de f por diferencias centradas.

4.1.6 Campo direcc.- Divide a la región en una malla de 24×24 y en cada uno de los nodos calcula y grafica el campo direccional. Usa el procedimiento F.

4.1.7 Lee.- Solicita al usuario el nombre del archivo donde se ha de guardar la pantalla actual añadiéndole la extensión .GRA Hace uso de los procedimientos PON_PETICION y TOMA_CADENA_CON_CURSOR.



5. Descripción

4.1 Gráfica:

Programa que resuelve el sistema de ecuaciones diferenciales y grafica su solución en la pantalla.

Parámetros de salida:

nombre - Nombre del archivo donde se guarda la gráfica
CADENA=STRING[80]

Procedimientos usados:

INICIO, VENTANA, LEEFUNCION, INTEGRA, CICLO, CAMPO_DIRECC,
LEE.

Ejemplo de llamada: grafica(nombre);

4.1.1 Inicio:

Contiene los datos iniciales.

Parámetros de salida:

x0,y0 - Coordenada de pantalla inicial del cursor.
INTEGER

x1,y1 - Coordenada izquierda superior de la pantalla.
REAL

x2,y2 - Coordenada derecha inferior de la pantalla.
REAL

xmin - Abcisa menor para la región de graficación.
REAL

xmax - Abcisa mayor para la región de graficación.
REAL

ymin - Ordenada menor para la región de graficación.
REAL

y_{max} - Ordenada mayor para la región de graficación.
REAL

título - Título inicial de la gráfica.
CADENA=STRING[80]

first - Bandera para iniciar la solución del sistema de ecuaciones.
BOOLEAN

tol - Tolerancia a satisfacerse en la solución otorgada por RKE.
REAL

thres - Tolerancia para controlar el error. Para uso de RKE.
VECTOR=ARRAY[-1,..4] OF REAL

Ejemplo de llamada: inicio(x₀,y₀,x₁,y₁,x₂,y₂,x_{min},y_{min},x_{max},y_{max},tol,
thres,first,título);

4.1.2 Ventana:

Reinicializa la región de graficación

Parámetros de salida:

**x_{min},y_{min},
x_{max},y_{max}** - Región de graficación.
REAL

título - Nuevo título de la gráfica.
CADENA=STRING[80]

first - Bandera para reiniciar la integración del sistema
de ecuaciones.
BOOLEAN

Procedimientos usados:

PON_PETICION, TOMA_CADENA_CON_CURSOR.

Ejemplo de llamada: ventana(x_{min},x_{max},y_{min},y_{max},título,first);

4.1.2.1 Pon_petición:

Procedimiento que pone un letrero en la pantalla.

Parámetros de entrada:

letrero	- Letrero a ser puesto en pantalla. STRING[80]
lin	- Renglón en el que se desea el letrero. INTEGER
col	- Columna en la que comienza el letrero. INTEGER

Ejemplo de llamada: pon_peticion('letrero',lin,col);

4.1.2.2 Toma_cadena_con_cursor:

Lee una cadena otorgada por el usuario en ambiente gráfico.

Parámetros de entrada:

lin, col	- Renglón y columna donde se localiza la cadena. INTEGER
longi	- Longitud deseada de la cadena. INTEGER
func	- Bandera que indica si se están leyendo las ecuaciones. BOOLEAN
num	- Cadena otorgada por el usuario. CADENA

Ejemplo de llamada: toma_cadena_con_cursor(num,23,74,5,func);

4.1.3 Leerfuncion:

Lee las ecuaciones de la pantalla; no las acepta hasta que no estén correctamente escritas.

Parámetro de salida:

func	- Indicador de que este procedimiento es exclusivamente para leer las ecuaciones. REAL
------	--

Procedimientos usados:

CONV, VENT_FUNCION.

Ejemplo de llamada: leefuncion(func);

4.1.3.1 Conv¹:

Revisa la sintaxis del sistema de ecuaciones

4.1.3.1.1 Leexp:

Lee las ecuaciones.

Parámetros de entrada:

reng - Renglón donde se encuentra la ecuación.
REAL

Parámetros de salida:

li - Cadena que contiene una de las ecuaciones del sistema.
CADENA

Procedimientos usados:

TOMA_CADENA_CON_CURSOR.

Ejemplo de llamada: leexp(li,reng);

4.1.3.2 Vent_funcion:

Coloca una ventana permanente en la región de graficación que contiene el sistema de ecuaciones actual.

Ejemplo de llamada: vent.funcion.

4.1.4 Integra:

Resuelve el sistema de ecuaciones y grafica en pantalla la solución.

Parámetros de entrada:

x - Tiempo inicial. En este caso estamos tomando $x = 0$.
REAL

xout - Tiempo final.
REAL

¹Elaborado por la Profesora Cecilia Ferra Oella de la Escuela de Matemáticas U.A.O.

h	- Paso de integración. REAL
y	- Condición inicial. VECTOR
first	- Variable lógica que especifica la primera y las subsecuentes llamadas a RKE. BOOLEAN
thres	- Error relativo deseado. VECTOR
tol	- Tolerancia absoluta deseada. REAL
llamada	- Bandera que indica la continuación de la graficación. BOOLEAN
fechas	- Indica si ha sido escogida una nueva condición inicial. BOOLEAN
limite	- Bandera para crear ambiente de graficación adecuado para usar CICLO. BOOLEAN
ciclo	- Bandera para usar CICLO. BOOLEAN

Parámetros de salida:

z	- Vector auxiliar donde se guarda el último tiempo calculado y el tiempo final. VECTOR
----------	--

Procedimientos usados:

CONDINI, REINICIO, RKE.

Ejemplo de llamada: `intgra(x,xout,first,z);`

4.1.5.1 Condini:

Lee las condiciones iniciales de la pantalla.

Parámetros de entrada:

- primer** - Indica el inicio o reinicio de la sesión. Permite parpadear el cursor.
BOOLEAN
- llmite** - Usado para desaparecer el cursor.
BOOLEAN

Parámetros de salida:

- flechas** - Para dar paso al movimiento del cursor.
BOOLEAN
- x0,y0** - Nuevas coordenadas de pantalla del cursor.
REAL
- xmund,ymund** - Nuevas coordenadas reales.
REAL
- ch** - Caracter que indica la tarea a realizar:
- F1** - Grafica con el tiempo hacia adelante
 - F2** - Grafica con el tiempo hacia atrás
 - F3** - Continúa graficando
 - F4** - Calcula y grafica campo direccional
 - F5** - Limpia la ventana actual de graficación
 - F6** - Imprime y guarda la gráfica
 - F7** - Guarda la gráfica en disco
 - F8** - Imprime la gráfica
 - ALT-F9** - Regresa al menú de graficación
 - F7** - Calcula y grafica el ciclo límite
 - ENTER** - Calcula y grafica el ciclo límite
 - ALT-F9** - Regresa al menú de graficación
 - F4** - Dibuja campo direccional
 - F5** - Limpia ventana
 - F8** - Solicita nueva región de graficación
 - F9** - Para cambiar el sistema de ecuaciones
 - F10** - Regresa al menú principal
- CHAR**

Procedimientos usados:

COORMUND, COMANDOS

Ejemplo de llamada: `condini(primer, limite, fecha, x0, y0, xmund, ymund, ch);`

4.1.5.1.1 Coordmund:

Efectúa la transformación de coordenadas de pantalla a coordenadas reales

Parámetros de entrada:

`x0, y0` - Coordenadas de pantalla del cursor.
INTEGER

Parámetros de salida:

`xmund, ymund` - Nuevas coordenadas reales.
REAL

Ejemplo de llamada: `coordmund(x0, y0, xmund, ymund);`

4.1.5.1.2 Comandos:

Pone en pantalla los comandos de graficación.

Parámetros de entrada:

`limite` - Elige si pone los comandos necesarios para CICLO.
BOOLEAN

`ch` - Para decidir la opción de campo direccional.
CHAR

Ejemplo de llamada: `comandos(limite, char);`

4.1.5.2 Reinicio:

Reinicia las condiciones iniciales.

Parámetros de entrada:

`ch` - Para indicar las condiciones iniciales a reinicializar.
CHAR

`xmund, ymund` - Última condición inicial.
REAL

`limite` - Para otorgar las condiciones iniciales referentes a CICLO.
BOOLEAN

atras	- Elige la dirección de integración. BOOLEAN
first	- Para iniciar nuevamente la integración. BOOLEAN
per	- Última aproximación al período. REAL
ord	- Última aproximación a la ordenada. REAL

Parámetros de salida:

y	- Condición inicial. VECTOR
z	- Almacena condiciones iniciales. VECTOR

Ejemplo de llamada: `reinicio(ch,xmund,ymund,per,ord, limite,atras,first,s,y);`

4.1.5.3 Rke:

Integra el sistema de ecuaciones diferenciales.

Parámetros de entrada:

x	- Tiempo inicial. Siempre tomamos $x = 0$ salvo cuando se desee continuar con la graficación, en cuyo caso se asigna $x := xout$. REAL
xout	- Tiempo final. REAL
h	- Paso de integración. REAL
y	- Condición inicial en x. VECTOR
first	- Variable lógica que especifica la primera y las subsecuentes llamadas a RKE. BOOLEAN

thres	- Error relativo deseado. VECTOR
tol	- Tolerancia absoluta deseada. Debe pertenecer al intervalo $[0.01 * eps, 10 * eps]$ donde <i>eps</i> es la unidad de redondeo de la máquina. REAL
llamada	- Bandera que indica la continuación de la graficación. BOOLEAN
fechas	- Indica si ha sido escogida una nueva condición inicial. BOOLEAN
lflmite	- Bandera para crear ambiente de graficación adecuado para usar CICLO . BOOLEAN
ciclos	- Bandera para usar CICLO . BOOLEAN

Parámetros de salida:

x	- Último valor del tiempo <i>x</i> . REAL
y	- Solución en <i>x</i> . VECTOR
h	- Paso tomado al integrar en <i>x</i> . REAL
lflag	- Bandera que nos indica si la integración tuvo éxito.
=0	la integración se llevó a cabo
=1	el número de pasos sobre un punto excedió a 100
=2	tol y thres resultan ser muy pequeñas
=-1	las condiciones iniciales son inadecuadas
	INTEGER

step - Longitud del paso.
REAL

ycoef - Vector que contiene los coeficientes para la interpolación de Hermite de 5º grado.
VECTOR

Procedimientos usados:

ESCOGEFUNCION.

Ejemplo de llamada: rke(x,y,h,first,iflag,step,ycoef,scr,ciclos,limite);

4.1.5.3.1 Escogefuncion:

Selecciona el sistema a resolver

Parámetros de entrada:

x - Punto en el que se va a evaluar.
REAL

y - Solución en el punto anterior.
VECTOR

ciclos - Selecciona el sistema referente a la localización de ciclos limite.
BOOLEAN

limite - Crea ambiente gráfico necesario.
BOOLEAN

Parámetros de salida:

yvalue - Evaluación en el nuevo punto.
VECTOR

Procedimientos usados:

EVALUAF.

Ejemplo de llamada: escogefuncion(xx,ynew,fout1,ciclos,limite);

4.1.5.3.1.1 Evaluaf⁽¹⁾:

Evalúa el lado derecho del sistema de ecuaciones en el punto dado.

4.1.5 Ciclo:

Calcula un punto y el período del ciclo límite.

Parámetros de entrada:

xmund, ymund	- Última condición inicial hasta antes de ser llamado este procedimiento. REAL
limite, atras, llamada, flechas, ciclo, func	- Para uso de INTEGRA. BOOLEAN
ch	- Para uso de REINICIO. CHAR

Parámetros de salida:

per	- Aproximación al período. REAL
ord	- Aproximación a la ordenada del punto. REAL

Procedimientos usados:

ELECCMAT, ALGORIT, INTEGRA.

Ejemplo de llamada: ciclo(xmund,ymun,ch,limite,atras,llamada,flechas,ciclos,func);

4.1.5.1 Eleccmat:

Elige la matriz inicial.

Parámetros de entrada:

xold	- Punto donde se va a aproximar el jacobiano. VECTOR
fold	- Vector que contiene la evaluación de la función f en xold. VECTOR

⁽¹⁾Elaborado por Cecilia Ferrer Colla. Escuela de Matemáticas UAC.

Parámetros de salida:

bb - Jacobiano inicial.
MATRIZ = ARRAY[1,..4,1..6] OF REAL

Procedimientos usados:

UNIDAD, EQUIS1, EQUIS2, DELTAS, W.W, EVALUAF, BINV.

Ejemplo de llamada: elecemat(xold,fold,bb);

4.1.5.1.1 Unidad:

Calcula la unidad de precisión de la computadora.

Parámetros de salida:

eps - Unidad de la computadora.
REAL

Ejemplo de llamada: unidad(eps);

4.1.5.1.2 Equis1:

Aproximación al primer punto

Parámetros de entrada:

yv - Primer punto.
VECTOR
eps - unidad de la máquina
REAL

Parámetros de salida:

yv1 - Aproximación al primer punto.
VECTOR

Ejemplo de llamada: equis1(yv,eps,yv1);

4.1.5.1.2 Equis2:

Aproximación al segundo punto

Parámetros de entrada:

yv	- Segundo punto. VECTOR
eps	- Unidad de la computadora. REAL

Parámetros de salida:

yv1	- Aproximación al primer punto. VECTOR
-----	---

Ejemplo de llamada: `equiv2(yv,eps,yv1);`

4.1.5.1.4 Deltas:

Calcula las variaciones para calcular el jacobiano inicial por diferencias.

Parámetros de entrada:

yv	- Punto en el cual se desea la aproximación. VECTOR
eps	- Unidad de la computadora. REAL

Parámetros de salida:

delta	- Variaciones. VECTOR
-------	--------------------------

Ejemplo de llamada: `deltas(yv,eps,delta);`

4.1.5.1.5 W_W:

Calcula la diferencia entre dos valores consecutivos de la función f .

Parámetros de entrada:

fold	- Penúltimo valor. VECTOR
ff	- Último valor de f . VECTOR

Parámetros de salida:

w - Diferencia f -fold.
VECTOR

Ejemplo de llamada: $W_W(\text{fold}, f, w)$;

4.1.5.1.7 Binvl:

Invierte la matriz original.

Parámetros de entrada:

bb - Matriz original.
MATRÍZ

Parámetros de salida:

bb - Matriz invertida.
MATRÍZ

Ejemplo de llamada: $\text{binv}(bb)$;

4.1.5.3 Algorit:

Calcula el nuevo punto.

Parámetros de entrada:

xold - Punto anterior.
VECTOR
p - Dirección de descenso.
VECTOR
alfa - Parámetro de búsqueda lineal.
REAL

Parámetros de salida:

x - Nueva aproximación.
VECTOR

Ejemplo de llamada: $\text{algorit}(xold, p, alfa, x)$;

Procedimientos usados:

NUEVALFA, DIRDESC.

4.1.5.3.1 Nuevaalfa:

Proporciona el parámetro de búsqueda lineal "alfa".

Parámetros de entrada:

alfapen - Penúltimo valor de alfa.
REAL

Parámetros de salida:

alfault - Valor de alfa actualizado.
REAL

Procedimientos usados:

FUNCIONORMA, PRODINT

Ejemplo de llamada: nuevaalfa(alfapen,alfault);

4.1.5.3.1.1 Funcionorma:

Calcula $f(x) = \frac{1}{2} \|F(x)\|^2$

Parámetros de entrada:

x - Punto en el que se va a efectuar el cálculo.
REAL
ff - Evaluación de $F(x)$.
VECTOR

Parámetros de salida:

norma - Valor de $f(x)$.
REAL

Procedimientos usados:

F_F

Ejemplo de llamada: funcionorma(x,ff,normien);

4.1.5.3.1.2 F_F:

Calcula $F(x)$

Parámetros de entrada:

xin - Punto en el que va a evaluar.
REAL

Parámetros de salida:

ff - Valor de $F(x)$.
VECTOR
ciclos - Para indicar la salida de CICLOS.
BOOLEAN

Procedimientos usados:

INTEG.APROX

Ejemplo de llamada: $f.f(xin,ff,ciclos)$;

4.1.5.2.1.1.1 Integ_aprox:

Integra el sistema en $[0, 1]$

Parámetros de entrada:

ttout - Extremo derecho donde se va a realizar la integración.
En este caso $ttout=1$.
REAL
limite - Para uso de RKE.
BOOLEAN
ciclos - Permite que se integre el sistema modificado.
BOOLEAN

Parámetros de salida:

xv - Solución.
VECTOR

Procedimientos usados:

RKE

Ejemplo de llamada: $integ_aprox(v,1,limite,ciclos)$;

4.1.5.2.1.2 Prodint:

Calcula $\nabla^T f(x)p$

Parámetros de entrada:

x - Punto en el que se va a ser el cálculo.
REAL
ff - Vector $F(x)$.
VECTOR
par - Parciales de f .
VECTOR
p - Dirección descendente en x .
VECTOR

Parámetros de salida:

prodint - Valor de $\nabla^T f(x)p$.
REAL

Procedimientos usados:

DIRDESC, PARCIALES.

Ejemplo de llamada: prodint(x,ff,par,p);

4.1.5.2.1.2.1 Dirdesc:

Calcula la dirección de descenso

Parámetros de entrada:

bb - Aproximación (invertida) del jacobiano.
MATRIZ
ff - valor de $f(x)$.
VECTOR

Parámetros de salida:

p - Dirección de descenso.
VECTOR

Ejemplo de llamada: dirdesc(bb,ff,p);

Procedimientos usados:

F_F, B_B

4.1.5.2.1.2.2 Parciales:

Calcula las parciales de f

Parámetros de entrada:

yv - Evaluación de f .
VECTOR

Parámetros de salida:

par - Parciales de f .
VECTOR

Procedimientos usados:

F_F

Ejemplo de llamada: parciales(yv,par);

4.1.5.2.2.1 B_B:

Siguiente aproximación al jacobiano.

Parámetros de entrada:

sbwstb - La matriz $(s_k - A_k w_k) s_k^T A_k$.
MATRIZ
stbw - Valor de $s_k^T A_k w_k$.
REAL
bb - Matriz A_k .
MATRIZ

Parámetros de salida:

bb - Matriz A_{k+1} .
MATRIZ

Procedimientos usados:

S_S, W_W, STBW, SBWSTB.

Ejemplo de llamada: b.b(sbwstb,stbw,bb);

4.1.6 Campo_direcc:

Calcula y grafica el campo direccional.

Procedimientos usados:

EVALUAF.

Ejemplo de llamada: campo.direcc

1.4.7 Lee:

Lee nombre de archivo para guardar gráfica.

Parámetros de entrada:

lin	- Renglón donde se pide el nombre. INTEGER
col	- Columna donde se pide el nombre. INTEGER
func	- Para uso de TOMA.CADENA.CON.CURSOR. BOOLEAN

Parámetros de salida:

nombre	- Nombre del archivo otorgado por el usuario más la extensión .GRA. CADENA
--------	---

Procedimientos usados:

PON_PETICION, TOMA.CADENA.CON.CURSOR.

Ejemplo de llamada: Lee(nombre,24,15,func);

5. Ejecución del sistema RF

Con el fin de mostrar el uso de RF, hemos optado por hacerlo mediante ejemplos intercalados entre las siguientes secciones.

5.1 El Péndulo.

La ecuación del péndulo

$$x'' + g \cdot \text{sen}x$$

puede ser reescrita como

$$x' = y$$

$$y' = -g \cdot \text{sen}x$$

Para iniciar la sesión, se debe teclear RF y oprimir la tecla [Enter], desplegándose en la pantalla el nombre y autores del programa RF. Al oprimir cualquier tecla aparece el menú siguiente:

MENU PRINCIPAL	
< 1 >	Graficar
< 2 >	Información
< 3 >	Desplegar gráficas
< 4 >	Terminar
	¿ Selección?

A continuación se describen cada una de las opciones.

< 1 > Graficar

Al seleccionar esta opción el programa solicita al usuario los valores x_{min} , x_{max} , y_{min} , y_{max} que determinan la región del plano en donde se va a graficar. Si estos valores no son proporcionados, lo cual ocurre al oprimir la tecla [Enter] o [Return], RF selecciona automáticamente los valores $x_{min} = -1$, $x_{max} = 1$, $y_{min} = -1$, $y_{max} = 1$. Tomemos nosotros

REGION DE GRAFICACION	
$x_{min} = -10$	$y_{min} = -10$
$x_{max} = 8$	$y_{max} = 8$

A continuación se solicita el título de la gráfica, el cual puede constar de hasta 70 caracteres. Después se solicitan las ecuaciones diferenciales de la manera siguiente, primero:

$$x' =$$

aquí se debe introducir la ecuación correspondiente y luego oprimir la tecla [Enter]. En la sección 4 se presenta una lista de comandos para la edición de las ecuaciones. El proceso se repite para la segunda ecuación. En nuestro ejemplo tendríamos

$x' = y$
$y' = -g \cdot \sin(x)$

En la pantalla aparecen los ejes coordenados, el título de la gráfica, las ecuaciones diferenciales en un recuadro, y lo más importante, un cursor en forma de cruz. El cursor puede ser movido a cualquier lugar de la pantalla y sirve para escoger las condiciones iniciales (x_0, y_0) , en el tiempo $t_0 = 0$, para el sistema de ecuaciones diferenciales.

Al mover el cursor aparece información adicional en la parte inferior de la pantalla (figura 1).

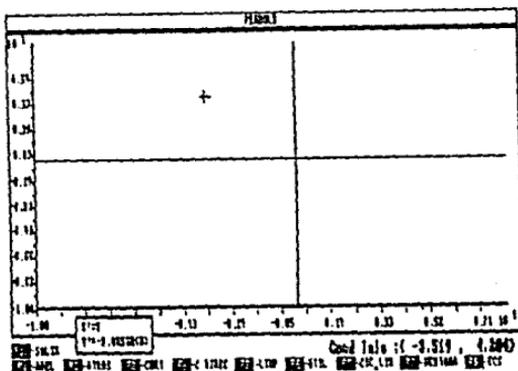


Fig. 1

Ahora estamos listos para graficar. Para esto se debe utilizar la tecla [F1] o la tecla [F2]. Al usar la tecla [F1], se resuelve el sistema con $t > 0$ y se grafica $x(t)$ vs $y(t)$. La tecla [F2] tiene una función similar, pero con $t < 0$ (Figuras 2 y 3).

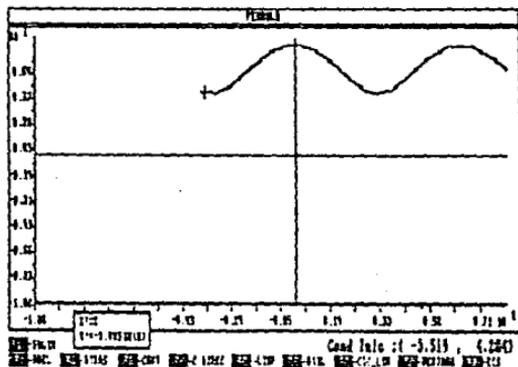


Fig. 2 Usando [F1].

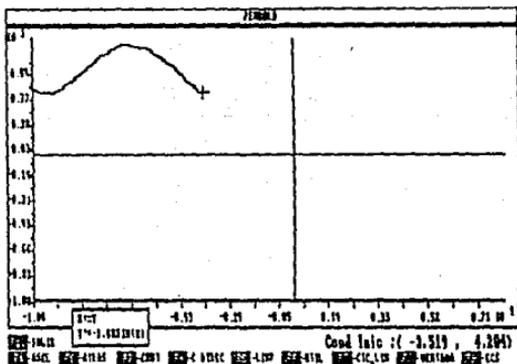


Fig. 5 Usando [F3].

Por supuesto es posible utilizar { F1 } y { F2 }, una después de la otra, como se muestra en la figura 4. Si después de usar una de las teclas anteriores se desea seguir graficando, el usuario debe usar la tecla { F3 }. Después de dibujar una trayectoria se puede mover el cursor a otra posición y repetir el proceso. La figura 5 ilustra lo anterior.

El usuario puede imprimir la gráfica usando la tecla { F6 } y siguiendo las instrucciones que el programa proporciona. Al oprimir { F6 } se tienen dos opciones, imprimir la gráfica o guardarla en disco para su uso posterior (ver la siguiente sección). Sólo es posible imprimir en impresoras compatibles con Epson de las series MX, RX y FX. En realidad no se imprime toda la información que aparece en la pantalla, solo aquella relevante como se muestra en la figura 6.

En este momento, si se desea, puede darse por terminada la sesión oprimiendo la tecla { F10 }. De manera automática RP guarda la última gráfica en un archivo llamado ULTIMA.GRA, la cual puede ser utilizada posteriormente como se verá más

adelante.

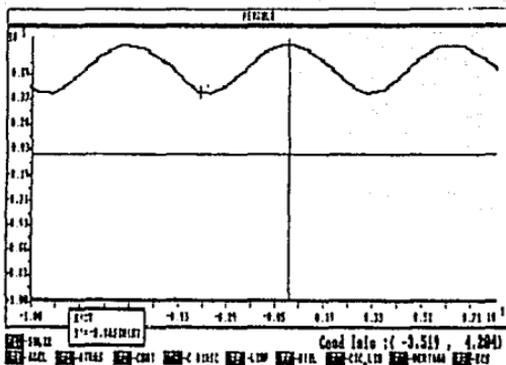


Fig. 4 Usando [F1] y [F8].

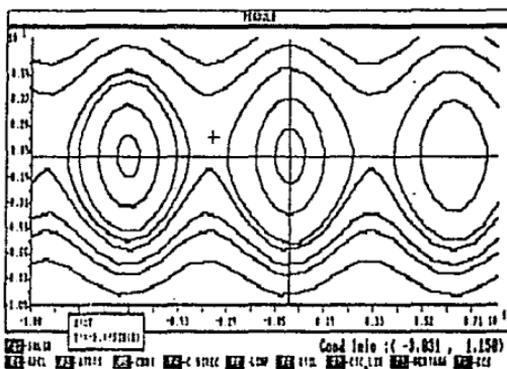


Fig. 5

< 2 > Información

Al escoger esta opción aparece en la pantalla, de forma resumida, el propósito del programa e información sobre los distintos comandos de RF.

< 3 > Despliega gráficas

Esta opción permite desplegar en pantalla gráficas almacenadas previamente en disco. Como se señaló anteriormente, siempre existe el archivo **ULTIMA.GRA** que contiene la última gráfica de la anterior sesión de trabajo. El usuario debe proporcionar el nombre del archivo conteniendo la gráfica a desplegar, teniendo cuidado en incluir la extensión **.GRA**. Una vez desplegada la gráfica, ésta puede ser impresa. Opcionalmente el usuario puede regresar al menú principal.

< 4 > Termina

Termina la sesión.

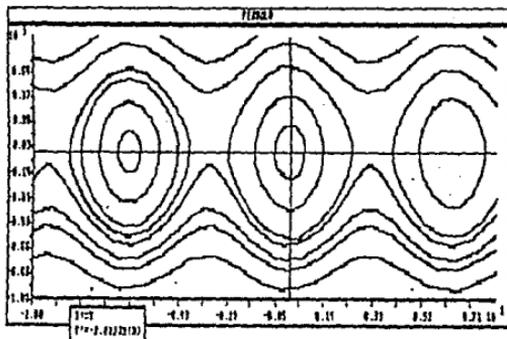


Fig. 6

5.2 Lista de comandos de graficación

Se presentan a continuación los comandos de graficación de **RF**.

F1 ADEL Gráfica con el tiempo avanzando hacia adelante.

F2 ATRAS Gráfica con el tiempo retrocediendo.

F3 CONT Continúa graficando la trayectoria.

F4 C.DIREC Despliega el campo direccional.

F6 LIMP Limpia la ventana de graficación.

F6 UTIL Utilerías. Aparecen las opciones siguientes:

F6 IMPR Imprime la gráfica.

F7 GUARDA GRAF Almacena la gráfica en disco.

ALT-F9 Regresa al menú de graficación.

F7 CIC_LIM Ciclo límite. Solamente que haya seguridad de la existencia de un ciclo límite se puede usar este comando (ver sección 6). Se escoge la condición inicial mediante el cursor. Aparecen los comandos siguientes:

F4 C DIREC Grafica el campo direccional.

F5 LIMP Limpia la ventana de graficación.

ENTER Calcula un ciclo límite.

ALT-F9 Regresa al menú de graficación.

F8 VENTANA Para cambiar z_{min} , z_{max} , y_{min} , y_{max} .

F9 ECS Nuevo sistema de ecuaciones.

F10 SALIR Abandona el ambiente gráfico y regresa al menú principal.

5.3 Comandos para la edición de funciones

Los comandos permitidos al momento de teclear las ecuaciones son:

+	Suma
-	Resta
*	Multiplicación
/	División
^	Potenciación, x^y es x^y
sin	Función seno
cos	Función coseno
tan	Función tangente
atan	Función arco tangente
ln	Función logaritmo natural
exp	Función exponencial(base e)
abs	Función valor absoluto
sqr	Función raíz cuadrada
max	$\max(f(x,y), g(x,y))$ es el máximo entre f y g
min	$\min(f(x,y), g(x,y))$ es el mínimo entre f y g

5.4 Resolviendo $y' = f(x, y)$

Como se mencionó en la introducción, es posible graficar soluciones de una ecuación diferencial de primer orden. Para esto se reescribe $f(x, y)$ como $f(x, y) = p(x, y)/q(x, y)$ y entonces se considera el sistema

$$\begin{aligned}x' &= q(x, y) \\ y' &= p(x, y)\end{aligned}$$

Considérese, por ejemplo, la ecuación $y' = x + x/y$. La manera más fácil es tomar $p(x, y) = f(x, y)$ y $q(x, y) = 1$, lo cual da

$$\begin{aligned}x' &= 1 \\ y' &= x + x/y.\end{aligned}$$

Otra posibilidad es $p(x, y) = xy + x$ y $q(x, y) = y$. Por lo tanto el sistema de ecuaciones sería

$$\begin{aligned}x' &= y \\ y' &= xy + x.\end{aligned}$$

La figura 7 muestra algunas soluciones del primer sistema de ecuaciones.

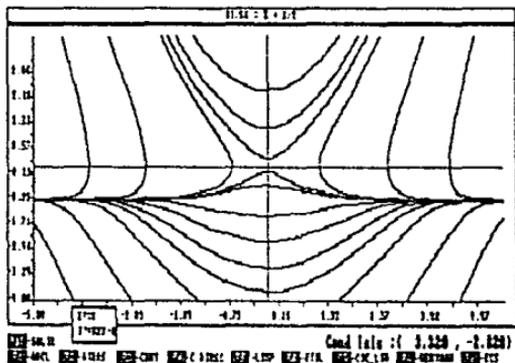


Fig. 7 Soluciones de $y' = x + x/y$.

5.5 Calculando ciclos límite

Una de las capacidades más importantes de BF es la localización de ciclos límite. Esto es posible cuando se conoce que ellos existen, o bien si se tiene

una fuerte evidencia de su existencia. Por ejemplo al graficar diversas trayectorias o al desplegar el campo direccional podría hacerse evidente la existencia de uno de estos ciclos.

La opción **F7 CIC_LIM** permite:

1. Localizar un punto sobre el ciclo límite y tomar a éste como condición inicial para graficar el ciclo.
2. Encontrar una aproximación al período del ciclo límite.

El usuario debe proporcionar una aproximación t_0 al período y un punto (x_0, y_0) con las características que se describen más adelante. Estas aproximaciones deben ser determinadas antes de oprimir la tecla [F7].

Aproximación al período

Para encontrar una aproximación al período t_0 , se debe escoger una condición inicial (x, y) cercana al lugar por donde se supone que pasa el ciclo límite y graficar la trayectoria $(x(t), y(t))$ con dicha condición inicial. Una aproximación t_0 al período será un valor de t tal que $(x(t), y(t))$ esté cercano a (x, y) , como se muestra en la figura 8. Así que el problema se reduce a localizar tal punto. Al graficar una trayectoria, en la parte inferior de la pantalla, aparece la opción **ESPACIO-PARAR**. Al oprimir la barra de espaciado del teclado la graficación es detenida. De esta manera, deteniendo la graficación en el momento oportuno, puede localizarse el punto $(x(t_0), y(t_0))$ deseado. **RF** toma de manera automática esta t_0 como una aproximación inicial al período. Se recomienda hacer varios ensayos para garantizar una buena aproximación a t_0 .

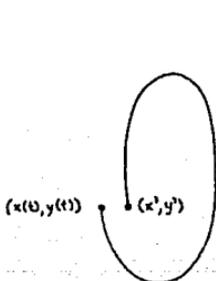


Fig. 8

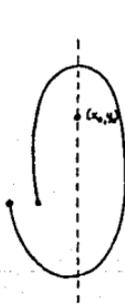


Fig. 9

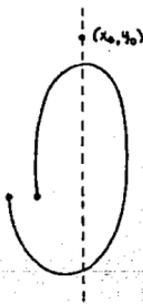


Fig. 10

5.4 Resolviendo $y' = f(x, y)$

Como se mencionó en la introducción, es posible graficar soluciones de una ecuación diferencial de primer orden. Para esto se reescribe $f(x, y)$ como $f(x, y) = p(x, y)/q(x, y)$ y entonces se considera el sistema

$$\begin{aligned}x' &= q(x, y) \\y' &= p(x, y)\end{aligned}$$

Considérese, por ejemplo, la ecuación $y' = x + x/y$. La manera más fácil es tomar $p(x, y) = f(x, y)$ y $q(x, y) = 1$, lo cual da

$$\begin{aligned}x' &= 1 \\y' &= x + x/y.\end{aligned}$$

Otra posibilidad es $p(x, y) = xy + x$ y $q(x, y) = y$. Por lo tanto el sistema de ecuaciones sería

$$\begin{aligned}x' &= y \\y' &= xy + x.\end{aligned}$$

La figura 7 muestra algunas soluciones del primer sistema de ecuaciones.

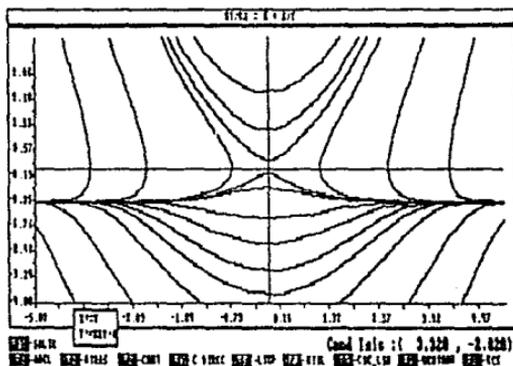


Fig. 7 Soluciones de $y' = x + x/y$.

5.5 Calculando ciclos límite

Una de las capacidades más importantes de RF es la localización de ciclos límite. Esto es posible cuando se conoce que ellos existen, o bien si se tiene

Aproximación a (x_0, y_0)

Después de determinar la aproximación al período, el cursor debe ser colocado sobre un punto (x_0, y_0) de forma tal que la recta $y = x_0$ interseque al ciclo límite (ver figuras 9 y 10).

En este momento ya se puede hacer uso de la opción [F7]. Al oprimir esta tecla, RF toma como condición inicial (x_0, y_0) , el punto donde se encuentra el cursor, y enseguida aparece un menú con las opciones siguientes:

ENTER > CICLO Inicia el cálculo del ciclo límite y de su período, t_0 es la t final de la última trayectoria graficada y (x_0, y_0) es tomado directamente del cursor.

F4-C DIREC Dibuja el campo direccional si así lo desea el usuario.

F5-LIMPIA Limpia la ventana de graficación.

ALT>F9 SALIR Regresa al ambiente gráfico anterior.

Para facilitar la comprensión de esta opción, se estudiará la ecuación de Van der Pol,

$$\begin{aligned}x' &= y \\ y' &= \epsilon y(1-x) - x\end{aligned}$$

con $\epsilon = 2.5$ y $x_{\min} = -5$, $x_{\max} = 5$, $y_{\min} = -5$, $y_{\max} = 5$. La figura 11 sugiere la presencia de un ciclo límite. Existen resultados teóricos que garantizan la existencia de tal ciclo.

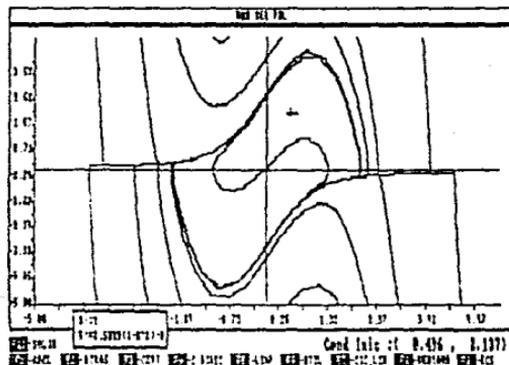


Fig. 11

Para encontrar una aproximación al período, se coloca el cursor sobre un punto con las características señaladas con anterioridad, por ejemplo en el punto $(-0.452, 1.886)$. Para encontrar una gráfica conveniente, en este ejemplo se usa [F1], luego [F3] y después la barra de espaciamento para detener la graficación en el momento adecuado, obteniéndose una gráfica como la que se muestra en la figura 12.

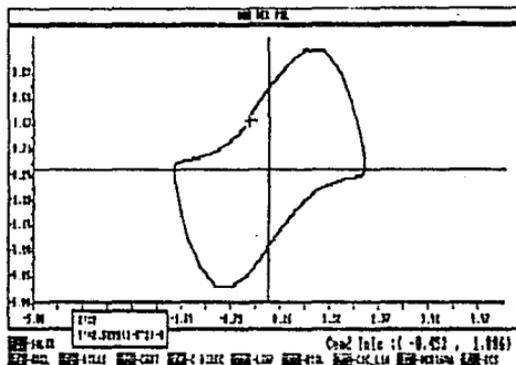


Fig. 12

Obsérvese que en este caso, el mismo punto donde se inició la trayectoria puede servir como una aproximación inicial (x_0, y_0) . Presionando [F7] y luego [Enter] se inicia la búsqueda del ciclo límite. Al término del proceso, el ciclo límite es graficado, el valor del período es desplegado y de forma automática RF regresa al ambiente de graficación previo.

CAPÍTULO VI

EJEMPLOS Y EXPERIENCIA NUMÉRICA

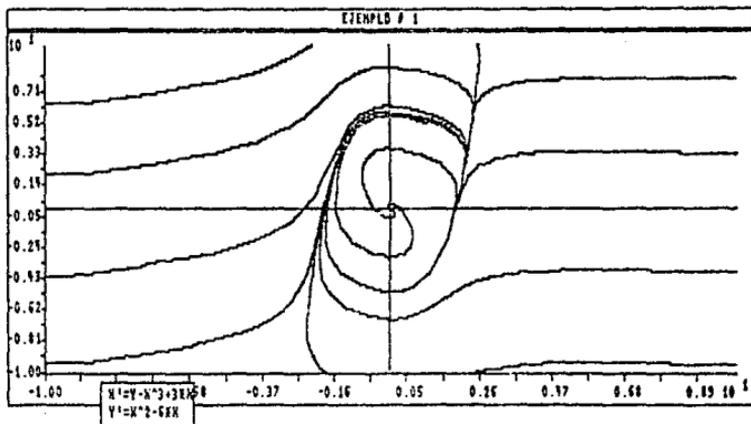
Con el objeto de validar el sistema RF hemos seleccionado una colección de ejemplos que se muestran a continuación.

En la localización de ciclos límite, es importante proporcionar una buena condición inicial para el período P_0 y la ordenada y_0 , que puede ser obtenida fácilmente mediante RF como se explica con detalle en V.5.

Como se mencionó en IV.3 el uso del mapeo de Poincaré como método para aproximar el período y un punto del ciclo límite existente, no es muy recomendable, pues requiere resolver la ecuación variacional asociada al sistema de ecuaciones diferenciales y además, debido a la naturaleza de RF, decidimos no hacer uso de él.

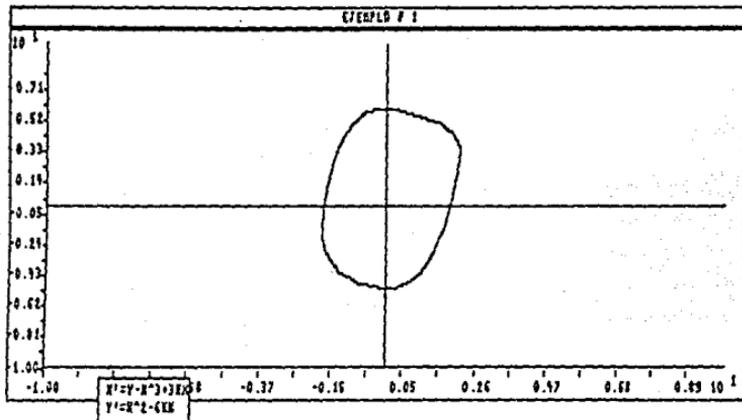
En el caso no-autónomo donde se sabe la existencia de un ciclo límite, nuestro problema se convierte en localizar solo el punto perteneciente a dicha órbita cerrada, pues el período es proporcionado por el sistema de ecuaciones. En una próxima versión tendremos implementado lo anterior.

Ejemplo 1



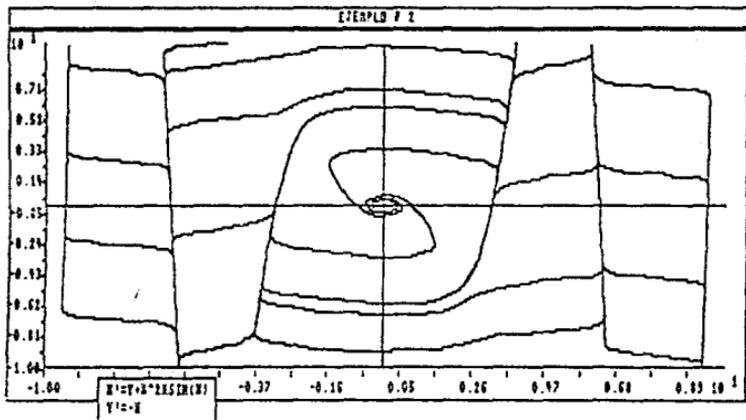
$$x' = y - x^3 + 3x$$

$$y' = x^2 - 6x$$



$P_0 = 2.92528$	$(x, y_0) = (2.074, 2.640)$
$P_1 = 2.97613$	$(x, y_1) = (2.074, 2.169)$
$P_2 = 2.97613$	$(x, y_2) = (2.074, 2.131)$
$P_3 = 2.97678$	$(x, y_3) = (2.074, 2.129)$
$P_4 = 2.97679$	$(x, y_4) = (2.074, 2.1288)$

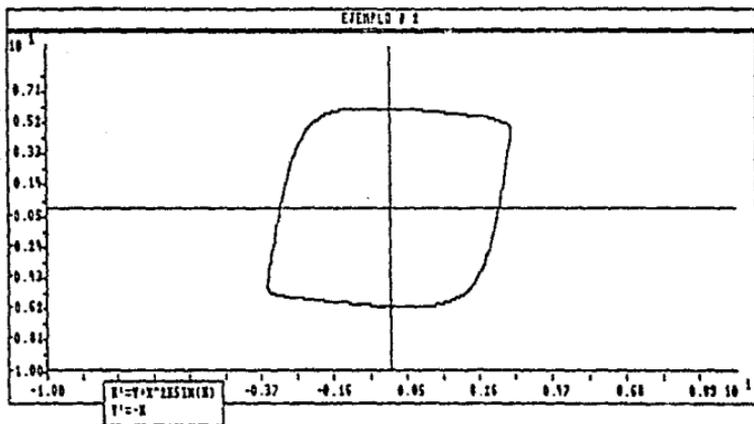
Ejemplo 2



$$x' = y + x^3 \sin(x)$$

$$y' = -x$$

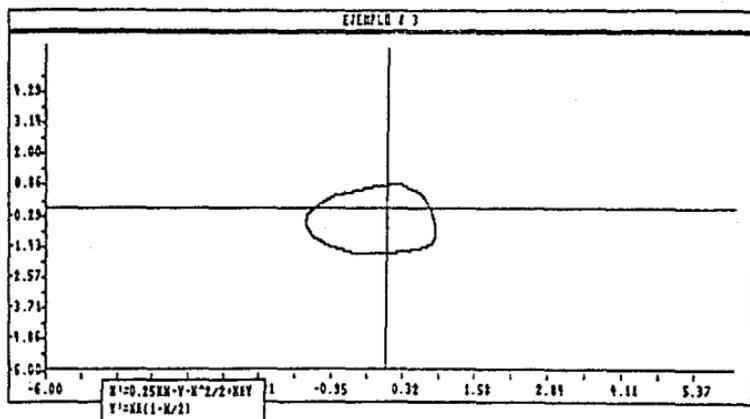
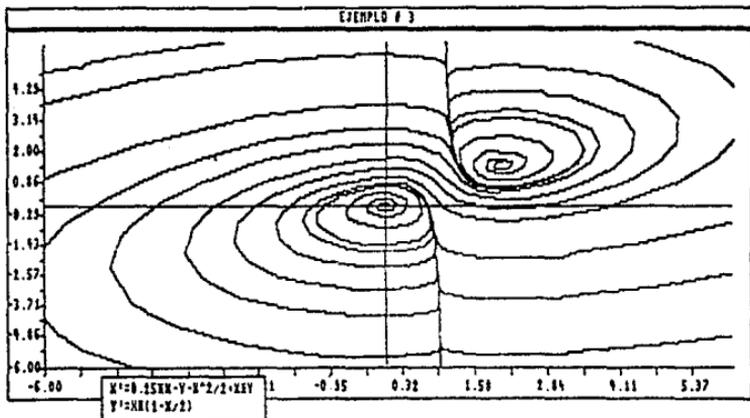
$P_0 = 9.01201$	$(x, y_0) = (3.428, 3.771)$
$P_1 = 9.01387$	$(x, y_1) = (3.428, 3.035)$
$P_2 = 9.01387$	$(x, y_2) = (3.428, 3.065)$
$P_3 = 9.01387$	$(x, y_3) = (3.428, 3.064)$
$P_4 = 9.01387$	$(x, y_4) = (3.428, 3.0642)$



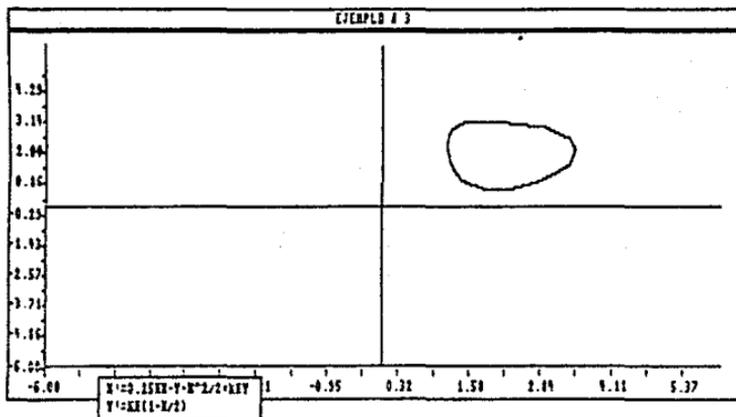
Ejemplo 3

$$x' = 0.25x - y - \frac{x^3}{2} + xy$$

$$y' = x(1 - \frac{x}{2})$$



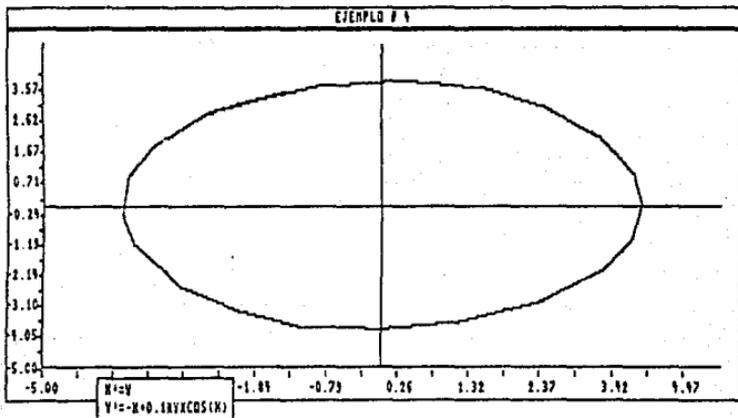
$P_0 = 8.11432$ $(x, y_0) = (0.757, -1.131)$
 $P_1 = 8.28887$ $(x, y_1) = (0.757, -1.438)$
 $P_2 = 8.26670$ $(x, y_2) = (0.757, -1.307)$
 $P_3 = 8.27233$ $(x, y_3) = (0.757, -1.349)$
 $P_4 = 8.27325$ $(x, y_4) = (0.757, -1.358)$
 $P_5 = 8.27314$ $(x, y_5) = (0.757, -1.357)$
 $P_6 = 8.27315$ $(x, y_6) = (0.757, -1.357)$
 $P_7 = 8.27315$ $(x, y_7) = (0.757, -1.3569)$



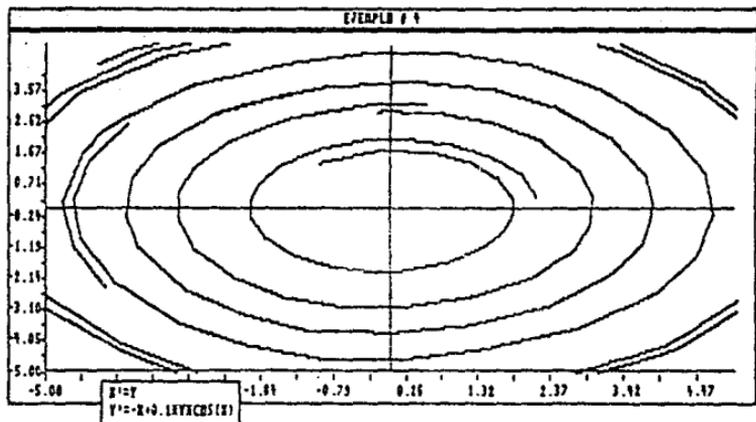
$P_0 = 8.43564$ $(x, y_0) = (1.245, 2.866)$
 $P_1 = 8.27410$ $(x, y_1) = (1.245, 2.862)$
 $P_2 = 8.27331$ $(x, y_2) = (1.245, 2.861)$
 $P_3 = 8.27316$ $(x, y_3) = (1.245, 2.861)$
 $P_4 = 8.27315$ $(x, y_4) = (1.245, 2.861)$
 $P_5 = 8.27315$ $(x, y_5) = (1.245, 2.8610)$

Ejemplo 4

$$\begin{aligned}x' &= y \\ y' &= -x + 0.1y \cos(x)\end{aligned}$$



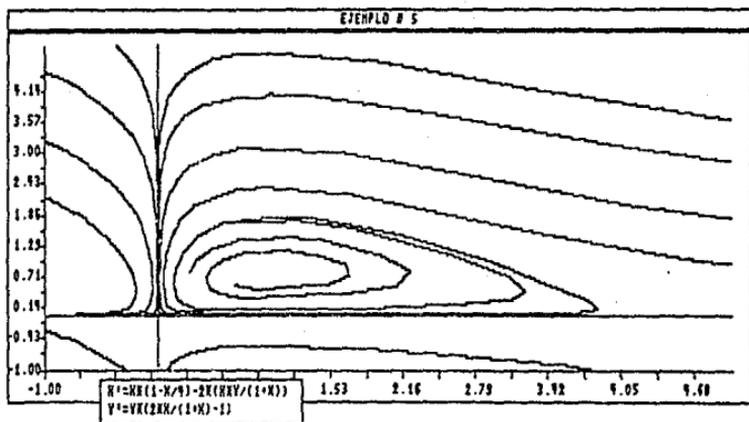
$$\begin{aligned}
 P_0 &= 6.27780 & (x, y_0) &= (-2.482, 2.891) \\
 P_1 &= 6.28500 & (x, y_1) &= (-2.482, 2.832) \\
 P_2 &= 6.28500 & (x, y_2) &= (-2.482, 2.8325)
 \end{aligned}$$



Ejemplo 5

En [14] se reporta la existencia de un ciclo límite para el sistema

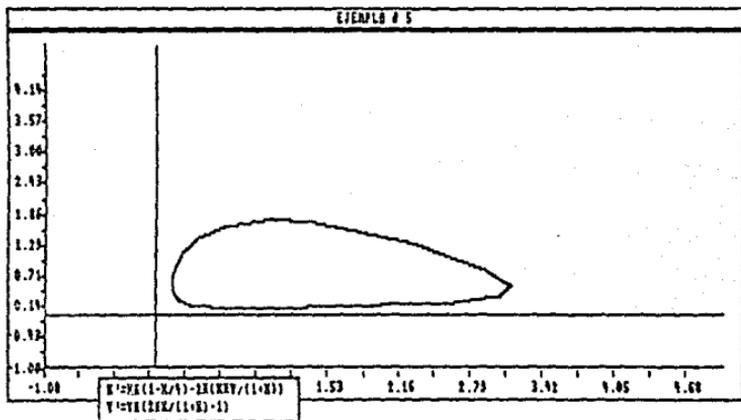
$$\begin{aligned}x' &= \gamma x \left(1 - \frac{x}{k}\right) - \left(\frac{m}{s}\right) \left(\frac{xy}{a+x}\right) \\y' &= y \left(\frac{mx}{a+x} - D_0\right)\end{aligned}$$



Con los parámetros $\gamma = 1, k = 4, m = 2, s = 1, a = 1, D_0 = 1$ obtuvimos los siguientes resultados

$$\begin{aligned}P_0 &= 12.90271 & (x, y_0) &= (0.186, 0.303) \\P_1 &= 13.072884 & (x, y_1) &= (0.186, 0.311) \\P_2 &= 13.07820 & (x, y_2) &= (0.186, 0.309)\end{aligned}$$

$$\begin{aligned}
 P_3 &= 13.07809 & (x, y_3) &= (0.186, 0.309) \\
 P_4 &= 13.07810 & (x, y_4) &= (0.186, 0.309) \\
 P_5 &= 13.07810 & (x, y_5) &= (0.186, 0.3090)
 \end{aligned}$$

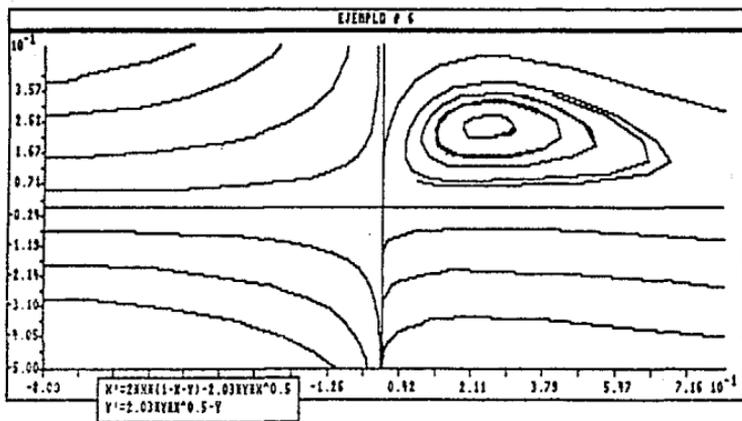


Ejemplo 6

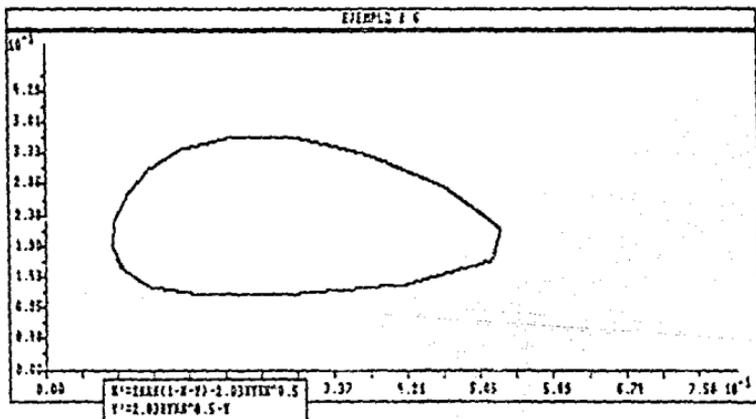
Un ciclo límite en un modelo depredador-presa [15]:

$$x' = 2x(1 - x - y) - 2.03yz^{\frac{1}{2}}$$

$$y' = 2.03yz^{\frac{1}{2}} - y$$



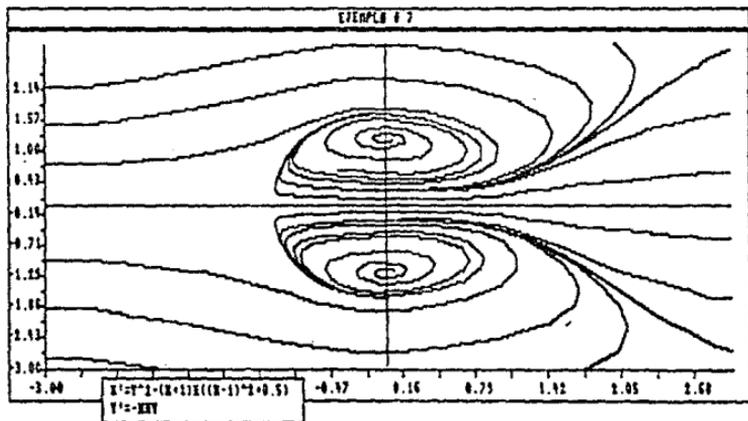
$P_0 = 7.96788$	$(x, y_0) = (0.079, 0.145)$
$P_1 = 7.81012$	$(x, y_1) = (0.079, 0.188)$
$P_2 = 7.82961$	$(x, y_2) = (0.079, 0.180)$
$P_3 = 7.72812$	$(x, y_3) = (0.079, 0.229)$
$P_4 = 7.77886$	$(x, y_4) = (0.079, 0.205)$
$P_5 = 7.80424$	$(x, y_5) = (0.079, 0.192)$
$P_6 = 7.81692$	$(x, y_6) = (0.079, 0.186)$
$P_7 = 7.81692$	$(x, y_7) = (0.079, 0.186)$
$P_8 = 7.84338$	$(x, y_8) = (0.079, 0.208)$
$P_9 = 7.85484$	$(x, y_9) = (0.079, 0.215)$
$P_{10} = 7.86068$	$(x, y_{10}) = (0.079, 0.220)$
$P_{11} = 7.87296$	$(x, y_{11}) = (0.079, 0.220)$
$P_{12} = 7.87341$	$(x, y_{12}) = (0.079, 0.221)$
$P_{13} = 7.87318$	$(x, y_{13}) = (0.079, 0.220)$
$P_{14} = 7.87334$	$(x, y_{14}) = (0.079, 0.220)$
$P_{15} = 7.87336$	$(x, y_{15}) = (0.079, 0.220)$
$P_{16} = 7.87336$	$(x, y_{16}) = (0.079, 0.2204)$



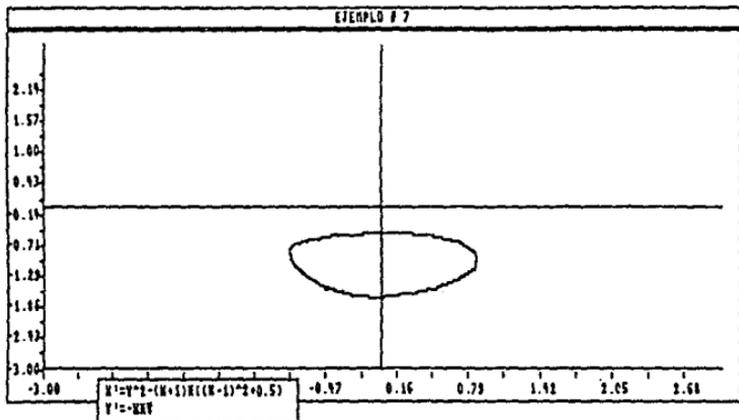
Ejemplo 7

Existen métodos para garantizar la existencia de ciclos límite de un sistema de ecuaciones diferenciales ordinarias. El método de comparación desarrollado en [10] nos proporciona la existencia de exactamente dos ciclos límite del sistema

$$\begin{aligned}x' &= y^2 - (x+1)((x-1)^2 + \mu) \\y' &= -xy\end{aligned}$$

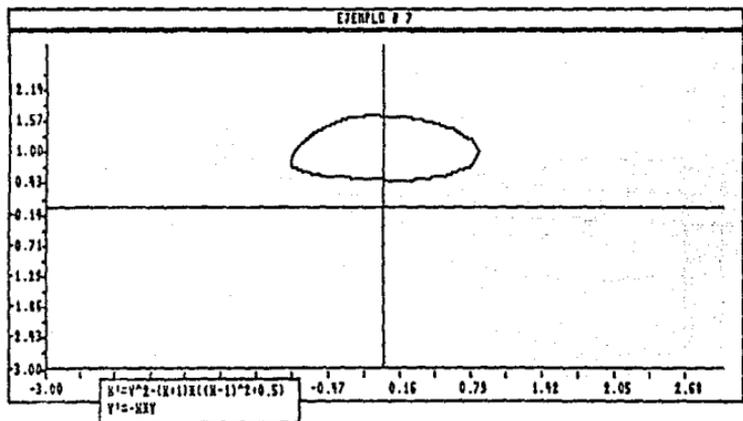


con $\mu = \frac{1}{2}$ localizamos los dos ciclos límite como se muestra enseguida



$P_0 = 4.48422$	$(x, y_0) = (-0.515, -1.433)$
$P_1 = 4.50612$	$(x, y_1) = (-0.515, -1.462)$
$P_2 = 4.50589$	$(x, y_2) = (-0.515, -1.462)$
$P_3 = 4.50589$	$(x, y_3) = (-0.515, -1.4621)$

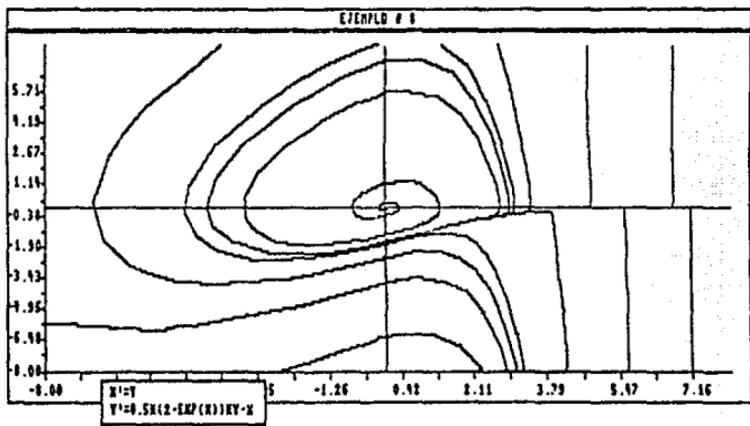
$P_0 = 4.38303$ $(x, y_0) = (-0.677, 1.282)$
 $P_1 = 4.50276$ $(x, y_1) = (-0.677, 1.252)$
 $P_2 = 4.50588$ $(x, y_2) = (-0.677, 1.255)$
 $P_3 = 4.50590$ $(x, y_3) = (-0.677, 1.255)$
 $P_4 = 4.50590$ $(x, y_4) = (-0.677, 1.2550)$



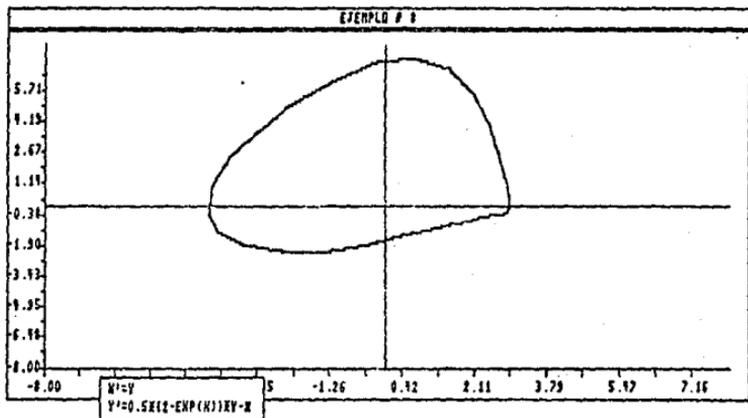
Ejemplo 8

$$x' = y$$

$$y' = \frac{1}{2}(2 - \exp(x))y - x$$



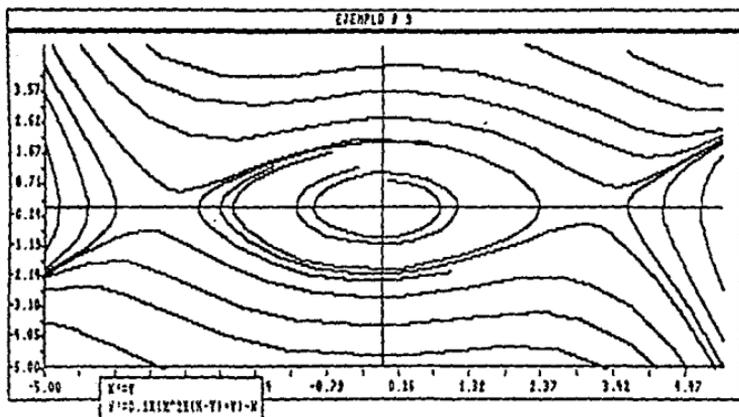
$P_0 = 8.18391$ $(x, y_0) = (2.742, 1.710)$
 $P_1 = 8.17436$ $(x, y_1) = (2.742, 1.768)$
 $P_2 = 8.17438$ $(x, v_2) = (2.742, 1.768)$
 $P_3 = 8.17438$ $(x, v_3) = (2.742, 1.7680)$



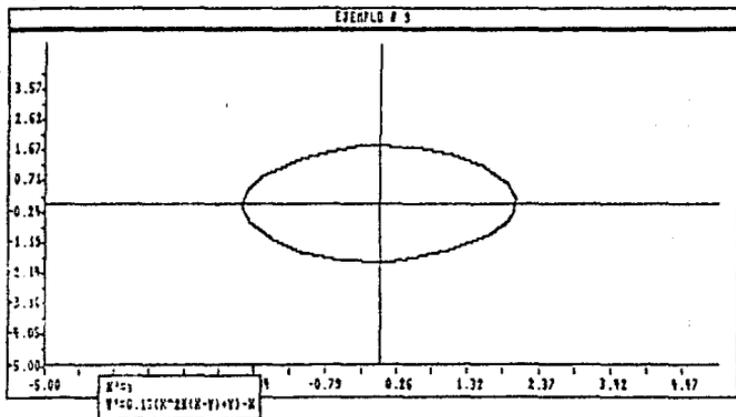
Ejemplo 9

$$x' = y$$

$$y' = \frac{1}{10}(x^2(x-y) + y) - x$$



$P_0 = 7.79281$	$(x, y_0) = (1.985, -0.440)$
$P_1 = 7.61733$	$(x, y_1) = (1.985, -0.309)$
$P_2 = 7.60893$	$(x, y_2) = (1.985, -0.343)$
$P_3 = 7.60956$	$(x, y_3) = (1.985, -0.342)$
$P_4 = 7.60979$	$(x, y_4) = (1.985, -0.342)$
$P_5 = 7.60988$	$(x, y_5) = (1.985, -0.343)$
$P_6 = 7.609888$	$(x, y_6) = (1.985, -0.3426)$

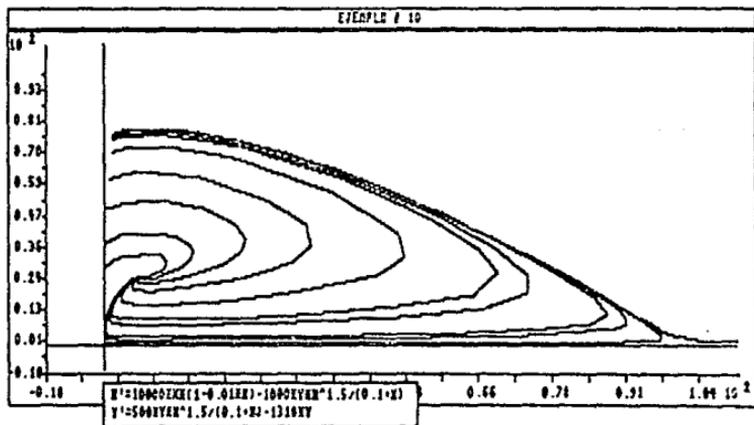


Ejemplo 10

Existen casos en que no se obtienen resultados satisfactorios como es el del siguiente sistema de ecuaciones [16] que describe la dinámica de un modelo depredador-presa, donde sólo se pudo constatar la existencia del ciclo límite como se muestra en el retrato fase.

$$\dot{x} = 10^4 x(1 - 0.1x) - \frac{10^3 y x^{\frac{3}{2}}}{0.1 + x}$$

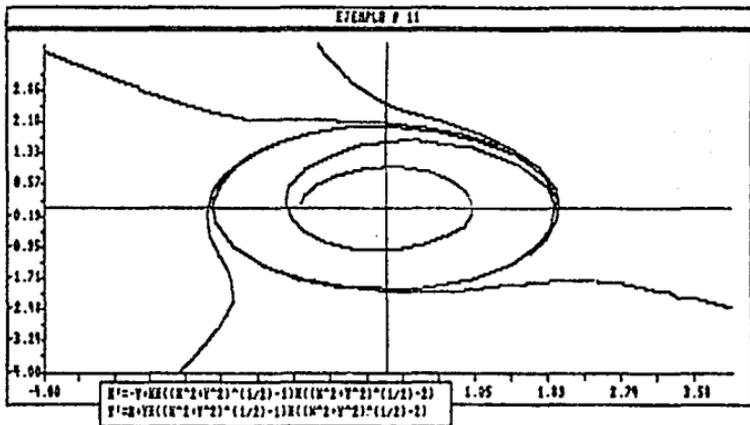
$$\dot{y} = \frac{500 y x^{\frac{3}{2}}}{(0.1 + x)} - 1310 y$$



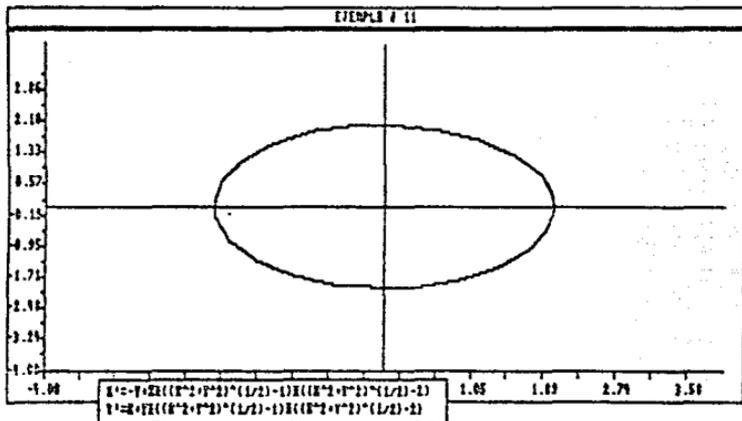
Ejemplo 11

$$x' = -y + x(\sqrt{x^2 + y^2} - 1)(\sqrt{x^2 + y^2} - 2)$$

$$y' = x + y(\sqrt{x^2 + y^2} - 1)(\sqrt{x^2 + y^2} - 2)$$



$$\begin{aligned}
 P_0 &= -6.21955 & (x, y_0) &= (1.913, 0.453) \\
 P_1 &= -6.28320 & (x, y_1) &= (1.913, 0.584) \\
 P_2 &= -6.28319 & (x, y_2) &= (1.913, 0.584) \\
 P_3 &= -6.28319 & (x, y_3) &= (1.913, 0.5845)
 \end{aligned}$$



BIBLIOGRAFIA

- [1] Allen Richard, Prues Stven, Shampine Laurence. *Fundamentals of Numerical Computing*. Publicación del Depto. de Matemáticas, Fac. de Ciencias UNAM.
- [2] Dennis J.E. Jr. Schnabel Robert B. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. Prentice-Hall.
- [3] Wilson H.K. *Ordinary Differential Equations*.
- [4] Hirsch Morris W., Smale Stephen. *Differential Equations, Dynamical Systems and Linear Algebra*. Editorial Alianza.
- [5] Kuo-Shung Cheng. *Uniqueness of a Limit Cycle for a Predator-Prey System*. SIAM J Math Anal Vol. 12 No.4 July 1981.
- [6] Hairer E. Norsett S.P. Wanner G. *Solving Ordinary Differential Equations I*. Editorial Springer-Verlag.
- [7] Shampine L.E. Watts H.A. *Comparing Error Estimators for Runge-Kutta Methods*. Math. of Comp. Vol 25, N° 115 Julio 1971.
- [8] Curry J. H. *An Algorithm for Finding Closed Orbits* Global thy of Dynamical Systems. Lecture Notes in Maths #819 Springer 1980.
- [9] Kubicek Milan and Holodniok Martin. *Algorithms for Determination Points in Ordinary Differential Equations*. Journal of Computational Physics 70, 203-217 (1987).
- [10] Ye Yan-Quian. *Theory of Limit Cycles* Translation of Mathematical Monographs. American Mathematical Society Vol 66.
- [11] Shampine Laurence F. Watts Herman A. *The Art of Writing a Runge-Kuta Code*. Part I pp 257-275. Mathematical Software III. Edit. J. R. Rice, Academic Press 1977.
- [12] Shampine Laurence F. Watts Herman A. *Practical Solution of Differential Ordinary Equations by Runge-Kutta Methods*. Sand 76-0585.
- [13] Hastings Alan. *Multiple Limit Cycles in Predator-Prey Models*. J. Math. Biology II, 51-63 (1981).
- [14] *Estimates of Predator-Prey Limit Cycles*. Bulletin of Mathematical Biology Vol. 37, 1975.