Revista Boliviana de Física 18, 55–59, 2011
ISSN 1562–3823. Indexada en: SciELO, Latindex, Periódica

ENSEÑANZA DE LA FÍSICA COMPUTACIONAL EN LA ESCUELA PROFESIONAL DE FÍSICA DE LA UNIVERSIDAD NACIONAL DE SAN AGUSTÍN DE AREQUIPA-PERÚ
COMPUTATIONAL PHYSICS TEACHING IN “LA ESCUELA PROFESIONAL DE FÍSICA DE LA UNIVERSIDAD NACIONAL DE SAN AGUSTÍN DE AREQUIPA-PERÚ”

Lic. Edwin Llamoca Requena†
Escuela profesional de Física de la
Universidad Nacional de San Agustín de Arequipa

RESUMEN

La enseñanza de la Física Computacional en la Escuela Profesional de Física de la Facultad de Ciencias Naturales y Formales de la Universidad Nacional de San Agustín (Perú) está orientada a que nuestros estudiantes se preparen para afrontar problemas complejos que se presentan en nuestra sociedad. Por tal motivo se han planificado tres cursos de Física Computacional. El primer curso comprende conceptos básicos del lenguaje de programación C y métodos numéricos básicos. Un segundo curso consiste en la aplicación de los métodos numéricos en la solución de las ecuaciones de la Física Matemática, para lo que se usa programación en C, Fortran, Matlab, Octave, Scilab y GNUPlot. El tercer curso está orientado a resolver problemas más complicados, debido a que se debe adquirir más experiencia en programación. El curso es asistido en forma personalizada, lo que quiere decir que todos los ejercicios se hacen en el centro de cómputo. Tal fue el impacto de este método de esnseñaza, que muchos estudiantes, se motivaron y pudieron publicar artículos relacionados con la dinámica molecular, dinámica no-lineal, métodos numéricos para el electromagnetismo. De acuerdo a la acreditación universitaria, el curso está orientado en plataforma Linux-Ubuntu, por lo que los trabajos se elaboran en Latex.

Descriptores: enseñanza de la física, computadoras en la enseñanza

Código(s) PACS: 01.40._d, 01.50.H_

ABSTRACT

The teaching Computational Physics in the Escuela Profesional de Física, Facultad de Ciencias Naturales y Formales of the Universidad Nacional de San Agustín (Perú) is oriented for preparing our students to be able to tackle complex problems that arise in our society. Three courses are planned for the subject of Computational Physics. The first course covers basic concepts of C programming language and basic numerical methods. A second course is based on the application of numerical methods in solving the equations of mathematical physics; the programming is made in C, Fortran, Matlab, Octave, Scilab and GNUPlot. The third course is designed to solve more complicated problems, in order to give the students more experience in programming. The course is attended in a personal way, which means that all the exercises are done in the datacenter. The impact of the course on the students was notorious and many of them were motivated and published articles related to molecular dynamics, nonlinear dynamics and numerical methods for electromagnetism. According to the university accreditation standards, the course is focused on the Linux-Ubuntu platform, thus the works are elaborated in Latex.

Subject headings: education in physics, computers in education

1. INTRODUCCI'ON

La Física Computacional (FC) es una modalidad de investigación en Física que se adiciona al método científico (Jong 1991). La enorme potencia computacional de que se dispone hoy en día nos permite simular mediante cálculos en un ordenador; el comportamiento de diversos tipos de sistemas físicos, lo que nos permite estudiarlos sin necesidad de realizar experimentos reales muy costosos y complicados, siendo a veces imposibles de realizar en la práctica, sino solamente mediante experimentos virtuales.

Para poder comprender los diversos tipos de sistemas físicos es necesario conocer un lenguaje de programación para luego saber programar cálculos

†Email: ellamoca@hotmail.com

numéricos y finalmente saber planificar y diseñar programas que simulen diversos tipos de sistemas físicos.

Para llegar a este objetivo general la Escuela Profesional de Física (EPF) en su plan curricular y de acuerdo a la acreditación universitaria; enseña tres cursos de FC en ambiente Linux-Ubuntu debido a que los lenguajes de programación como C, Fortran y como Octave, Scilab y GNUPlot son software libre.

2. FÍSICA COMPUTACIONAL I

Para desarrollar este curso se deberá tener aprobado un primer curso de matemáticas y física. El objetivo principal será programar en lenguaje de programación de nivel intermedio C en forma básica y posteriormente aplicarlos a los métodos numéricos. A la vez se dictan dos clases básicas de Latex para que los estudiantes presenten sus trabajos en formato PDF.

Los métodos numéricos básicos que se programan son (Chapra & Canale 2008; Chainskaia & Doig 1999; Gerald & Heatley 2000):

- Método de Newton,
- Método de bisección,
- Método de la secante,
- Método de falsa posición,
- Interpolación de Lagrange,
- Método de diferencias finitas,
- Método del trapecio,
- Método de Simpson,
- Método de Gauss,
- Método de Gauss Siedel,
- Método de Euler,
- Método de Heun y
- Métodos de Runge-Kutta.

Todas las prácticas se hacen en el centro de cómputo con asistencia personalizada debido a que cada estudiante tenga confianza en programar y perder el miedo al ordenador. En cada práctica se ponen variantes al método y solo ellos tendrán que modificar el código.

3. FÍSICA COMPUTACIONAL II

Con conocimientos básicos del lenguaje de programación C, ondas y óptica y métodos matemáticos de la física I; orientamos el curso en su interpretación física de los códigos en C, Fortran, Octave, Scilab y GNUPlot y relacionar con procesos físicos.

Esta es una tarea my ardua por parte del profesor para poder relacionar códigos en diferentes lenguajes de programación, software de aplicaciones y relacionar con sistemas físicos que se quiere simular.

Como se sabe la sintaxis del lenguaje de programación C, es muy detallada y extensa; entonces, explicar Fortran, Octave o GNUPlot se hace mucho más sencilla y los estudiantes captan con rapidez la sintaxis y por tanto la programación se hace más fácil.

Cada una de las prácticas que a continuación se detallan; se realizan en el centro de cómputo:

- Oscilador armónico simple,
- Oscilador armónico amortiguado,
- Oscilador armónico forzado,
- Integración por Montecarlo (Jong 1991),
- Problema de dos cuerpos (Jong 1991),
- Problema de n cuerpos (Jong 1991),
- Ecuación logística (Jong 1991),
- Ecuaciones de Lorenz (Chapra & Canale 2008; Jong 1991),
- Secciones de Poincaré (Jong 1991),
- Ecuación de Laplace (Mathews & Fink 2000),
- Ecuación de onda (Mathews & Fink 2000),
- Ecuación del calor (Mathews & Fink 2000) y
- Elemento finito (Chapra & Canale 2008; Gerald & Heatley 2000).

Las prácticas en el que hay ecuaciones diferenciales, se implementan los códigos con el método de Euler, pidiendo como ejercicio modificar con el método de Runge-Kutta de orden 2, 3 y 4; para establecer las diferencias respectivas en forma gráfica.

Se incide en forma permanente las gráficas en el diagrama de fases, para entender las secciones de Poincaré (Jong 1991).

Para las prácticas referentes a las ecuaciones en derivadas parciales, se resuelven por el método de diferencias finitas tomando en cuenta las condiciones a la frontera de Dirichlet y de Newmann (Chapra & Canale 2008; Mathews & Fink 2000; Gerald & Heatley 2000; Llamoca 2010). Se dan tentativas para ecuaciones más complejas como las ecuaciones de Poisson, ecuación de onda amortiguada. También la posibilidad de extender las ecuaciones a 2 dimensiones.

La aplicación del método Montecarlo en resolver integrales tiene muchas variedades y un alcance a aplicaciones físicas (Jong 1991).

Las ecuaciones de Lorenz, para saber la sensibilidad de las condiciones iniciales (Chapra & Canale 2008; Jong 1991).

La aplicación del elemento finito para una barra calentada en estado estable; es una práctica típica para entender la física del problema (Chapra &

Canale 2008; Gerald & Heatley 2000). Su programación es más avanzada pero bien que ya a estas alturas del curso era de esperarse.

Hasta aquí el nivel del curso está orientado a que el estudiante este preparado a afrontar sus futuros trabajos de investigación.

El alcance en este curso, es que al estudiante se le da como tarea programar en Octave o en C y Fortran.

4. FÍSICA COMPUTACIONAL III

Este último curso tiene como objetivo orientar al estudiante a usar los métodos de la física computacional para desarrollar sus futuros trabajos de investigación.

Reforzar las líneas de investigación que desarrolla la EPF con métodos de FC; es también el fin que tiene este curso.

Las líneas de investigación que desarrolla la EPF son: Física Médica, Física Teórica, Termoluminiscencia, Energías renovables, Películas Delgadas y Pedagogía de la Enseñanza de la Física.

Entonces se ha planificado el curso orientado a reforzar estas líneas de investigación de tal forma que él estudiante este bien identificado con alguna de dichas líneas.

Aquí la programación la fábrica el mismo estudiante con la tutoría del profesor de curso. Es decir, el profesor explicará al detalle la física del problema con indicaciones de que método numérico se tiene aplicar para resolver un sistema físico.

Las prácticas han sido bien seleccionadas tomando en cuenta la mayoría de las líneas de investigación.

La práctica interacción radiación materia aplicando el método de Montecarlo es una identificación muy precisa para resolver problemas más complejos que se presentan en la física médica (Llamoca 2010, 2000).

La práctica de resolver la ecuación diferencial parcial parabólica por el método theta y por el método de Montecarlo, es el inicio para tratar problemas más complejos de las Energías renovables (Chapra & Canale 2008; Mathews & Fink 2000; Gerald & Heatley 2000; Llamoca 2010).

Una tercera práctica es resolver el modelo de Ising en 2 dimensiones, el cual es un problema clásico de física estadística. Práctica muy interesante en el que se da el inicio de resolución de problemas sistemas complejos.

La cuarta práctica será resolver el autómata celular y fractales, que relacionan problemas complejos de la física teórica (Plaza IMCA; Hearn & Baker 1995).

La quinta práctica, sobre procesamiento de señales (Epstein 2003; Gonzáles & Woods 1996) en el que se aplican FFT (Umez-Eronini 2001) y Convolución (Oppenheim & Willsky 1994; Oppenheim & Schafer 2000) que tiene mucha aplicación en física médica.

Como última práctica se ha previsto incluir en este curso, algoritmos genéticos y lógica difusa para el tratamiento de sistemas complejos.

Se ha querido tratar otros temas de la FC como el elemento finito en 2 dimensiones (Gerald & Heatley 2000; Chandrupatla & Belegundu 1999), pero tiene ya tiene una complejidad en elaborar los códigos.

Bueno, hay muchos temas que se pueden abordar en la FC, pero eso será parte de futuros temas de tesis para bachillerato y licenciatura.

El impacto de esta metodología de la enseñanza de la FC fue muy relevante porque en tres años consecutivos se publicaron trabajos relacionados con la FC. Trabajos como Dinámica Molecular, Dinámica no lineal, Física Médica. A continuación se presenta una práctica tipo que motiva a seguir con la FC.

5. OSCILADOR DE DUFFING

Una buena parte de la introducción al caos se basa en un modelo mecánico llamado oscilador de Duffing (Jong 1991) cuya ecuación diferencial que gobierna es no lineal, es decir:

$$\frac{d^2x}{dt^2} + c\frac{dx}{dt} - x + x^3 = f\cos\omega t \tag{1}$$

Para dar solución a esta ecuación se aplica el método de Runge-Kutta de orden 4, es decir:

$$\begin{aligned} x_{n+1} &= x_n + h\upsilon_n + \frac{h}{6}\left(k_1 + k_2 + k_3\right) \\ \upsilon_{n+1} &= \upsilon_n + \frac{1}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right) \\ t_{n+1} &= t_n + h \end{aligned}$$

Para:

$$\begin{aligned} k_1 &= ha\left(t, x, \upsilon\right) \\ k_2 &= ha\left(t + \frac{h}{2}, x + \frac{h\upsilon}{2}, \upsilon + \frac{k_2}{2}\right) \\ k_3 &= ha\left(t + \frac{h}{2}, x + \frac{h\upsilon}{2} + \frac{hk_1}{4}, \upsilon + \frac{k_2}{2}\right) \\ k_4 &= ha\left(t + h, x + h\upsilon + \frac{hk_2}{2}, \upsilon + k_3\right) \end{aligned}$$

Donde se relaciona con la ecuación (1) de la siguiente forma:

$$a\left(t, x, \upsilon\right) = -c\upsilon + x - x^3 + f\cos\omega t$$

Que sigue siendo la ecuación (1).

La tendencia de cualquier oscilación, es que tiende hacia un punto o puntos el cual se llama atractor. Tratándose de este tipo de dinámica el atractor será un conjunto de puntos, que será un atractor caótico.

Para fabricar dicho atractor, se aplica la sección de Poincaré, el cual se define tiempos circulares, donde cada tiempo circular el periodo.

$$t = \frac{2\pi}{\omega} \tag{2}$$

Esto se debe a que para un tiempo muy largo el sistema se adecuará a la frecuencia ω corresponde a la fuerza externa. Si se divide el tamaño de paso h como:

$$h = \frac{t}{m} = \frac{2\pi}{m\omega} \tag{3}$$

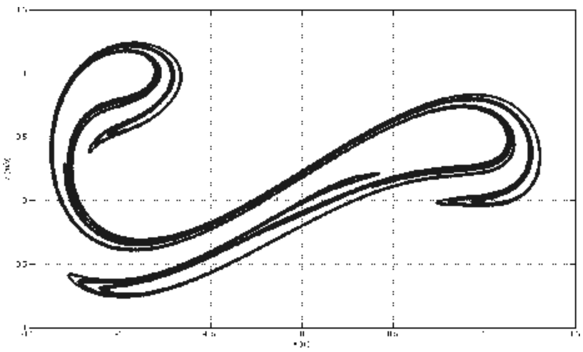

FIG. 1.— Atractor caótico.

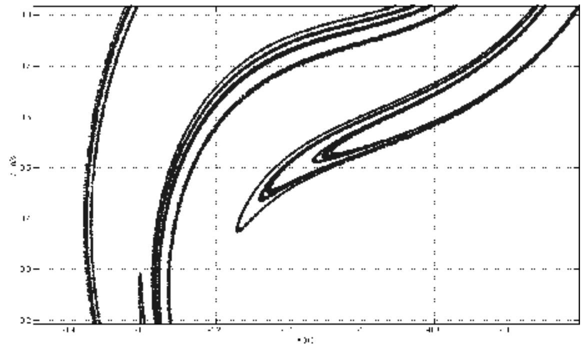

FIG. 2.— Primer acercamiento.

es decir cada período se divide en m partes, de tal manera que si el programa empieza a correr desde un $t = 0$, y cuando se haya evaluado el método Runge-Kutta, m veces, significa que se ha cumplido un periodo y en ese momento del programa grafica en el diagrama de fases (x, v). Por cada otro periodo también se podrá graficar otro (x, v), y así sucesivamente.

Con estas indicaciones el programa en Octave es:

```
clear, clf, hold off
n=0; h=0.05; m=20;
% Constantes del programa
c=0.24; b=1; d=1; f=0.68; w=1.7;
h=2*pi/(w*m);
% Condiciones Iniciales
t=0; x=1; v=1; tfin=1000000;
% Inicio de la Simulacion
pt(1)=t; pv(1)=v; px(1)=x;
while t<tfin
   n=n+1;
   for i=1:m
       a=feval('df',t,x,v,c,f,w);
       k1=h*a;
       t1=t+h/2; x1=x+h*v/2; v1=v+k1/2;
       a=feval('df',t1,x1,v1,c,f,w);
       k2=h*a;
       x2=x1+h*k1/4; v2=v+k2/2;
        a=feval('df',t1,x2,v2,c,f,w);
        k3=h*a;
        t3=t+h; x3=x1+h*k2/4; v3=v+k3;
        a=feval('df',t3,x3,v3,c,f,w);
        k4=h*a;
        x=x+h*v+h*(k1+k2+k3)/6;
        v=v+(k1+2*k2+2*k3+k4)/6;
        t=t+h;
        if x>+pi
           x=x-2*pi;
        end
        if x<-pi
           x=x+2*pi;
        end
    end
    px(n+1)=x; pv(n+1)=v;
  end
  plot(px,pv,'.'); grid on;
  xlabel('x(m)'), ylabel('v (m/s)');
```

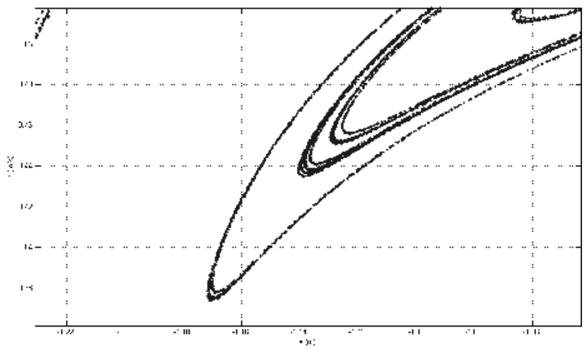

FIG. 3.— Segundo acercamiento.

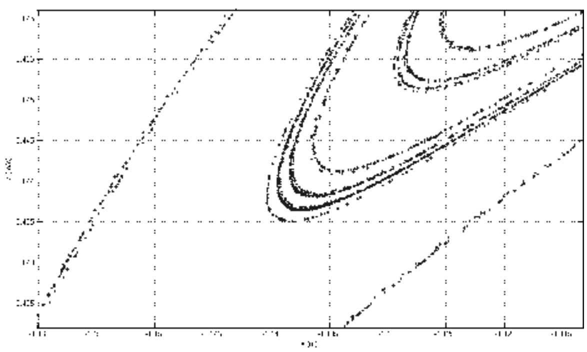

FIG. 4.— Tercer Acercamiento.

Cuya gráfica se muestra en la figura 1, en el que se muestra un atractor caótico de naturaleza fractal para el oscilador de Duffing. La situación fractal se manifiesta acercando ciertas zonas y se verá un conjunto de puntos alineados y si se acerca más habrá otro conjunto de puntos alineados; es decir, es un atractor caótico de naturaleza fractal (Jong 1991). Ver las figuras 2, 3 y 4.

La belleza de estas figuras motiva mucho a un estudiante. Pero la motivación crece, cuando se le pide buscar otro tipo de atractores con el mismo oscilador de Duffing, o con otros osciladores no lineales.

Se deja como ejercicio pasar este código a C y graficar en GNUPlot. Como también averiguar las secciones de Poincaré para un oscilador armónico simple, amortiguado y forzado.

6. BIBLIOGRAFÍA

La bibliografía para los cursos de Física Computacional es muy amplia.

Para Física Computacional I, el libro guía es: Chapra & Canale 2008. Otros libros de consulta son: Gould & Tobochnik 1988; Chainskaia & Doig 1999; Mathews & Fink 2000; Gerald & Heatley 2000.

Para Física Computacional II, los libros guía son: Chapra & Canale 2008; Jong 1991; Mathews & Fink 2000; Llamoca 2010.

Otros libros de consulta son: Gould & Tobochnik 1988; Farlow 1982; Nakamura 1997; Perez 2003; dle Río & Cabezas 2002; Gerald & Heatley 2000; Marshall 1985; Chandrupatla & Belegundu 1999; Hearn & Baker 1995.

Para Física Computacional III, los libros guía son: Chapra & Canale 2008; Mathews & Fink 2000; Oppenheim & Willsky 1994; Llamoca 2010.

Otros libros de consulta son: Gould & Tobochnik 1988; Farlow 1982; Plaza IMCA; Marshall 1985; Epstein 2003; Umez-Eronini 2001; Gonzáles & Woods 1996; Chandrupatla & Belegundu 1999; Llamoca 2000.

7. CONCLUSIONES

Con esta metodología de enseñanza de la Física Computacional resultó positiva porque la mayoría de los estudiantes que llevaron el curso quedaron satisfechos.

Con esta forma de enseñanza, los estudiantes pueden leer diferentes lenguajes de programación e interpretar físicamente.

Con esta metodología, los estudiantes con mucha seguridad, pueden adaptarse a otros software especializados, como Comsol, Femlab, MCNP, Penélope etc.

Todos los estudiantes de este curso, presentan sus trabajos en Latex, para presentarlos en formato PDF.

REFERENCIAS

Chainskaia, L. & Doig, E. 1999, Elementos de Anaálisis Numérico (Pontifica Universidad Católica del Perú)

Chandrupatla, T. R. & Belegundu, A. D. 1999, Introducción al Estudio del Elemento Finito en Ingeniería (Prentice Hall)

Chapra, S. C. & Canale, R. P. 2008, Métodos Numéricos para Ingenieros (Mc. Graw Hill)

dle Río, J. A. I. & Cabezas, J. M. R. 2002, Métodos Numéricos (Pirámide)

Epstein, C. L. 2003, Introduction to the Mathematics of Medical Imaging (Prentice Hall)

Farlow, S. J. 1982, Partial Differential Equations for Scientists and Engineers (John Wiley & Sons)

Gerald, C. F. & Heatley, P. O. 2000, Análisis Numérico con Aplicaciones, Sexta Edición (Prentice Hall)

Gonzáles, R. C. & Woods, R. E. 1996, Tratamiento Digital de Imágenes (Addison-Wesley)

Gould, H. & Tobochnik, J. 1988, Computer Simulation Methods (Addison-Wesley Publishing Company)

Hearn, D. & Baker, M. P. 1995, Gráficas por Computadora (Prentice Hall)

Jong, M. L. D. 1991, Introduction to Computational Physics (Addison-Wesley Publishing Company)

Llamoca, E. 2000, Tesis: Energía absorbida en phantom proveniente de una fuente puntual de Ra-226 aplicando el Método de Monte Carlo

—. 2010, Física Computacional (Apuntes de Clase)

Marshall, G. 1985, Solución Numérica de Ecuaciones Diferenciales, tomo 2: Ecuaciones en Derivadas Parciales (Editorial Reverté Argentina)

Mathews, J. H. & Fink, K. D. 2000, Métodos Numéricos con MatLab, Tercera Edición (Prentice Hall)

Nakamura, S. 1997, Análisis Numérico y Visualización Gráfica con MatLab (Pearson Educación)

Oppenheim, A. V. & Schafer, R. W. 2000, Tratamiento de Señales en Tiempo Discreto, Segunda Edición (Prentice Hall)

Oppenheim, A. V. & Willsky, A. S. 1994, Señales y Sistemas (Prentice Hall)

Perez, C. 2003, MatLab y sus aplicaciones en las ciencias y la Ingeniería (Pearson Prentice Hall)

Plaza, S. IMCA, Fractales y Genración Computacional de Imágenes

Umez-Eronini, E. 2001, Dinámica de Sistemas y Control (Thomson Learning)