

Plan de Estudio Doctorado en Ingeniería

Mención: Mecánica Computacional

MATEMÁTICA APLICADA

Objetivos

El objetivo central del curso es poner en contacto a los alumnos con las ecuaciones diferenciales, que constituyen el tipo de ecuación fundamental en que resultan numerosos modelos de la Física, Química, Biología, etc. Se pretende que los alumnos conozcan el comportamiento cualitativo de las soluciones de las ecuaciones diferenciales, tanto ordinarias, como en derivadas parciales, siendo capaces de determinarlo dependiendo del tipo de ecuación (elíptica, parabólica, hiperbólica). También se pretende que los alumnos se familiaricen con algunos métodos analíticos de resolución, que en algunas circunstancias permiten hallar formas cerradas de las mismas, y en otras permiten obtener conclusiones acerca de su comportamiento cualitativo. El programa que se propone está diseñado de modo que modelización, teoría y métodos de resolución participen de manera balanceada en el desarrollo del curso.

Programa sintético

Repaso de los teoremas de la divergencia y del rotor y rudimentos de la teoría de potencial. Modelos matemáticos. Leyes de conservación. Relaciones constitutivas. Transporte. Difusión. La ecuación del calor y la de Laplace. La ecuación de ondas. Ecuaciones en derivadas parciales de primer orden. Aplicaciones. La ecuación del calor. Problemas con condición inicial y condiciones de borde. Separación de variables. Principio de máximo y unicidad. Condiciones de borde independientes del tiempo. Estado estacionario. Condiciones de borde dependientes del tiempo, el Teorema de Duhamel. Series de Fourier. Transformada de Fourier. Transformada de Laplace. Aplicaciones a EDP en dominios infinitos. Problema de Sturm-Liouville. Separación de variables para ecuaciones del calor, Laplace y ondas en dimensiones superiores. Funciones especiales. Coordenadas generalizadas. Función de Green.

Conocimientos previos requeridos

Matemática de grado.

Carga horaria: Teoría: 60 horas. Coloquio y/o práctica en aula o laboratorio: 30 horas. Total: 90 horas.

Duración: 15 semanas.

Formas de evaluación: trabajos prácticos, 2 exámenes parciales y examen final.

Bibliografía básica

Arfken G.B., Weber H.J., "Mathematical Methods For Physicists", HARCOUT-Academic Press, 2001.

- Bleecker D., Csordas G., "Basic Partial Differential Equations", International Press, Cambridge, Massachusetts, 1996.
- Courant R., Hilbert D., "Methods of Mathematical Physics", Vols. I y II, John Wiley and Sons, 1953.
- Haberman R., "Elementary Applied Partial Differential Equations", Prentice Hall, Upper Saddle Rver, NJ, 1998.
- Larsson S., Thomée V., "Partial Differential Equations with Numerical Methods", Springer, 2009.
- Logan J. D. "Applied Partial Differential Equations", Springer, New York, 2004.

MECÁNICA DE FLUIDOS

Objetivos

El curso está orientado a proveer una base común de transferencia de cantidad de movimiento para alumnos provenientes de distintas carreras de ingeniería. El núcleo del curso reside en el estudio de flujos de fluidos Newtonianos. La ecuación de Navier- Stokes es analizada en detalle, en particular, los casos límite de bajos y altos números de Reynolds que se traducen en flujos reptantes por un lado, y por el otro, en flujo potencial (solución externa) y capa límite (solución interna). Se enfatizan los principios gobernantes más que la práctica ingenieril, sin embargo, también se hace uso de balances macroscópicos para obtener soluciones aproximadas.

Programa sintético

Principios de mecánica. Sistemas aislados y no aislados. Sistemas no inerciales. Estática de fluidos. Tensión superficial. Ecuaciones de transporte. Teorema del transporte. Ecuaciones diferenciales e integrales. Tensor de tensiones y tensor de deformaciones. Ecuación del movimiento. Fluido Newtoniano. Ecuación de Navier Stokes. Vorticidad. Flujo irrotacional. Ecuaciones de la energía y entropía. Adimensionalización de las ecuaciones de Navier Stokes. Casos particulares y números adimensionales significativos. Flujos viscosos y flujos a altos números de Reynolds. Flujos unidireccionales. Soluciones de similaridad. Flujos cuasi unidireccionales. Flujos alrededor de cuerpos sumergidos. Capa límite laminar. Solución de Blasius. Método de Karman y Pholhausen. Capa límite turbulenta. Ondas superficiales. Teoría de ondas de pequeña amplitud. Trayectoria de partículas. Aproximación de aguas poco profundas.

Conocimientos previos requeridos

Mecánica de fluidos de grado.

Carga horaria: Teoría: 60 horas. Coloquio y/o práctica en aula o laboratorio: 30 horas. Total: 90 horas.

Duración: 15 semanas.

Formas de evaluación: Trabajos prácticos, 1 examen parcial y examen final.

Bibliografía básica

Batchelor G. K., "An Introduction to Fluid Dynamics", Cambridge University Press, 1972.

Schlichting H., "Boundary layer Theory", Mc. Graw - Hill, 1979.

Slattery J. C., "Momentum, Energy and Mass Transfer in Continua", Mc Graw- Hill, 1972.

Whitaker S., "Introduction to Fluid Mechanics", Krieger Publishing Company, Prentice Hall, 1968.

White F. M., "Viscous - Fluid Flow", Mc Graw- Hill, 1974.

MECÁNICA DE SÓLIDOS

Objetivos

Introducción a los elementos principales de la teoría matemática de Elasticidad no lineal, aplicación de esta teoría a la solución de problemas de valores de frontera en sólidos no lineales, y análisis de las propiedades mecánicas de materiales sólidos bajo grandes deformaciones.

Programa sintético

Tensores. Tensores Cartesianos. Álgebra tensorial. Campos tensoriales. Análisis de Deformación y Movimiento. Cinemática. Deformación. Análisis del movimiento. Objetividad. Ecuaciones de Equilibrio. Conservación de la cantidad de movimiento. Tensor de Cauchy. Punto de vista Lagrangiano. Tensores de tensión conjugados.

Elasticidad. Leyes constitutivas para materiales simples. Material elástico de Cauchy. Materiales de Green. Problemas de Valores de Frontera. Formulación. Teorema de Ericksen. Algunas soluciones. Principios variacionales. Método de los Elementos Finitos en Elasticidad No Lineal. Discretización de las ecuaciones de equilibrio. Formulación Galerkin. Elementos finitos tetraédricos lineales. Aplicaciones.

Conocimientos previos requeridos

Ninguno.

Carga horaria: Teoría: 45 horas. Coloquio y/o Práctica en el aula o laboratorio: 30 horas. Total: 75 horas.

Duración: 15 semanas.

Formas de evaluación: 1 examen parcial, 1 trabajo final y examen final.

Bibliografía básica

Azaro R. J. y Lubarda V. A., "Mechanics of Solid Materials", Cambridge University Press, 2006.

Bathe K. J., "Finite Element Procedures", Prentice Hall, 1996.

de Souza Neto E. et al, "Computational Methods for Plasticity", J. Wiley & Sons, 2008.

Gurtin M. E., "An Introduction to Continuum Mechanics", Academic Press, 1981.

Ogden R. W., "Nonlinear Elastic Deformations", John Wiley & Sons, 1984.

Oliver X. y Agelet de Saracibar C., "Mecánica de Medios Continuos para Ingenieros", UPC Ediciones, 2002.

INTRODUCCIÓN AL MÉTODO DE LOS ELEMENTOS FINITOS

Objetivos

Conocer las bases matemáticas del método de los elementos finitos para problemas de campos escalares (térmicos, difusión, flujo potencial) y vectoriales (ecuaciones de elasticidad), así como comprender los aspectos prácticos de programación involucrados en el mismo.

Programa sintético

Introducción al MEF para problemas elípticos. Formulación variacional para un problema modelo unidimensional. MEF para problema modelo con funciones lineales por tramos. Estimación de error para MEF para el problema modelo. MEF para la ecuación de Poisson. Espacios de Hilbert. Interpretación geométrica del MEF. Problema de Neumann. Condiciones de borde naturales y esenciales. Formulación abstracta del MEF para problemas elípticos. Problema continuo. Discretización. Estimación de error. Norma energía. Ejemplos. Algunos espacios de elementos finitos. Requerimientos de regularidad. Ejemplos de elementos finitos.

Teoría de aproximación para el MEF. Estimaciones de error para problemas elípticos. Interpolación con funciones lineales por tramos en dos dimensiones. Interpolación con polinomios de grado superior. Estimaciones de error para el MEF en problemas elípticos. Regularidad de la solución exacta. Métodos adaptativos. Una estimación de error en norma L_2 . Aplicaciones para problemas elípticos. Problema de elasticidad. Problema de Stokes. Problema de flexión de placas. Elementos finitos curvos e integración numérica. MEF para problemas parabólicos. Problema modelo unidimensional. Semidiscretización en el espacio. Discretización en espacio y tiempo. Métodos de diferencias hacia atrás de Euler y Crank-Nicolson. Método de Galerkin discontinuo. Estimaciones de error, control automático del paso de tiempo y del paso espacial. Problemas hiperbólicos. Problema de convección-difusión. Métodos numéricos para problemas hiperbólicos. Método de Galerkin estándar. Difusión artificial clásica. Método de difusión por líneas de corriente. Método de Galerkin discontinuo. Sistemas de Friedrichs. Problemas no lineales.

Conocimientos previos requeridos

Álgebra matricial. Conocimientos básicos de computación y mecánica del continuo.

Carga horaria: Teoría. 60 horas. Coloquio y/o Práctica en el aula o laboratorio: 15 horas. Total: 75 horas

Duración: 15 semanas.

Formas de evaluación: Trabajos prácticos, 1 examen parcial y examen final.

Bibliografía básica

Bathe K.J. y Wilson E.L., "Numerical methods in finite element analysis", Prentice-Hall, 1976.

Hughes T.J.R., "The finite element method", Prentice-Hall Int. Editions, 1987.
Johnson C., "Numerical solution of partial differential equations by the finite element method", Cambridge University Press, 1995.
Zienkiewicz O.C. y Taylor R.L., "The finite element method", 5th ed. Butterworth-Heinemann, 2000.

MECÁNICA RACIONAL

Objetivos

Introducción a los elementos principales de la teoría de la mecánica del movimiento de sistemas de partículas, cuerpos rígidos y sistemas de cuerpos rígidos. Aplicación de esta teoría a la solución de problemas de mecanismos. Revisión de métodos numéricos necesarios a la solución de este tipo de problemas. Experimentación numérica con programas de elementos finitos para solución de estos problemas.

Programa sintético

Ecuaciones del Movimiento. Coordenadas generalizadas. Principio de mínima acción. Relatividad de Galileo. Lagrangiano de una partícula. Lagrangiano de un sistema de partículas. Leyes de Conservación. Energía. Cantidad de movimiento. Centro de masa. Cantidad de movimiento angular. Integración de las Ecuaciones de Movimiento. Movimiento en una dimensión. Masa reducida. Movimiento en un campo central. Problema de Kepler. Oscilaciones de Pequeña Amplitud. Oscilaciones libres en una dimensión. Oscilaciones forzadas. Oscilaciones de sistemas de más de un grado de libertad. Oscilaciones amortiguadas. Oscilaciones forzadas con fricción. Resonancia paramétrica. Oscilaciones no lineales. Resonancia con oscilaciones no lineales. Cinemática de Movimientos Finitos. Representación matricial. Descripción cinemática del movimiento de cuerpo rígido. Análisis de velocidades. Análisis de aceleraciones. Movimiento esférico infinitesimal. Parametrización de rotaciones. Vector de rotación. Forma de Cayley de la matriz de rotación. Parámetros de Rodríguez. Parámetros de Euler. Álgebra de cuaterniones. Vector de rotación conforme. Angulos de Euler. Movimiento de un cuerpo rígido. Velocidad angular. Tensor de inercia. Cantidad de movimiento angular del cuerpo rígido. Ecuaciones de movimiento del cuerpo rígido. Ecuaciones de Euler. Cuerpos rígidos en contacto. Movimiento en un marco de referencia no-inercial. Sistemas multicuerpos. Restricciones cinemáticas holonómicas y no holonómicas. Solución numérica de problemas algebraicos con restricciones. Problemas dinámicos con restricciones. Clasificación de pares cinemáticos. Uniones más comunes: rotoidal, prismática, cilíndrica, tornillo, plana, etc.

Conocimientos previos requeridos

Conocimientos de álgebra y ecuaciones diferenciales ordinarias.

Carga horaria: Teoría: 45 horas. Coloquio y/o práctica en aula o laboratorio: 30 horas. Total: 75 horas.

Duración: 15 semanas.

Formas de evaluación: Trabajos prácticos, 1 examen parcial y examen final.

Bibliografía básica

Landau L.D. y Lifshitz E. M., "Mechanics", 3rd edition, Pergamon Press, 1976.

Goldstein H., "Classical Mechanics". Addison Wesley, 1964.
Gérardin M. y Cardona A., "Flexible Multibody Dynamics, A Finite Element Approach", J. Wiley, 2001.
SAMTECH Group, "Samcef Mecano - User Manual", Liege, 2002.

MODELOS CONSTITUTIVOS PARA MATERIALES DISIPATIVOS: APLICACIÓN A MECÁNICA DE SÓLIDOS

Objetivos

El curso se orienta a brindar al alumno nociones básicas del modelado fenomenológico del comportamiento de materiales, como constituyentes de medios sólidos, bajo estados de carga. Especial énfasis se dará al aspecto computacional y de falla de materiales. Se pretende que el alumno consiga desarrollar la capacidad de modelar numéricamente los diversos comportamientos introducidos en el curso.

Programa sintético

Principios fundamentales en el modelado constitutivo de materiales: objetividad y simetrías. Termodinámica del continuo: balance de energía, desigualdad de entropía y disipación, variables internas para sistemas disipativos, potenciales termodinámicos, transformación de Legendre. Elasticidad: diferentes descripciones, hiperelasticidad y energía de deformación. Plasticidad: base termodinámica, superficie de fluencia, ecuaciones constitutivas para flujo asociado, Principio de Máxima Disipación, relaciones tangentes para superficies de fluencia suaves, modelo específico de plasticidad (Teoría J2), propiedades espectrales del tensor constitutivo tangente, criterio de estabilidad del material y unicidad. Viscoplasticidad: formulación de Perzyna, formulación de Duvaut-Lions. Daño: aspectos fenomenológicos, variable de daño, formulación termodinámica, daño isotrópico, modelos de plasticidad acoplado con daño. Aplicaciones al modelado numérico de falla de materiales: inestabilidad del material y bifurcación inducida por la misma; regularización del comportamientos constitutivo; modelos de elementos finitos aplicados a esta temática.

Conocimientos previos requeridos

Mecánica del continuo, programación en Matlab.

Carga horaria: Teoría: 30 horas. Coloquio y/o práctica en aula o laboratorio: 30 horas. Total: 60 horas.

Duración: 12 semanas.

Formas de evaluación: Trabajos prácticos y examen final.

Bibliografía básica

Runesson K., "Constitutive Theory and computational techniques for dissipative materials, Part II", Notas del curso ofrecido en la UPC, Barcelona, 1999.
Gurtin M.E., "An introduction to continuum mechanics", Academic Press, Florida, 1981.
Simo J.C. y Hughes T.J.R., "Elastoplasticity and viscoplasticity: computational

aspects”, Springer-Verlag, 1995.

Lemaitre J. y Chaboche J. L., “Mechanics of solid materials”, Cambridge Press, 1994.

Nguyen Q. S., “Stability and nonlinear solid mehanics”, John Wiley & Sons, 2000.

CÁLCULO CIENTÍFICO CON COMPUTADORAS PARALELAS

Objetivos

El objetivo del curso es brindar una introducción al uso de computadoras paralelas en cálculos científicos. Se considera la codificación o adaptación de algoritmos para procesamiento en paralelo, teniendo en cuenta las diferentes arquitecturas de computadoras paralelas. Se hace énfasis en el trabajo tanto en redes de computadoras (computadoras personales o estaciones de trabajo) como en los denominados clusters de procesadores (grupos de microprocesadores interconectados que pueden ser utilizados como una computadora paralela), ambas opciones disponibles en la Universidad Nacional del Litoral. Se prevén aplicaciones orientadas a problemas de la mecánica estructural y de fluidos.

Programa sintético

Introducción. Motivación. Desafíos actuales en necesidad de cálculo científico. Evolución de las supercomputadoras. Cálculo paralelo. Arquitectura de computadoras paralelas. Criterios de clasificación y taxonomía. Medidas de velocidad de procesamiento. Velocidad de procesamiento. Patrones de comparación (benchmarks). Optimización. Vectorización. Procesadores vectoriales. Programación de computadoras vectoriales. Análisis de algoritmos. Grafos. Análisis de algoritmos secuenciales. Análisis de algoritmos paralelos. Modelos y estrategias de paralelización. Niveles de paralelismo. Modelos de programación paralela. Estrategias de paralelización. Reordenamiento de las variables. Descomposición del dominio. Problemas de proyección en computadoras masivas. Diseño de programas paralelos. Programación en el modelo de memoria compartida. Procesos. Mecanismos de coordinación. Dependencia entre los datos. Granularidad y equilibrio de las tareas. Programación en paralelo en el paradigma de memoria compartida. HPF (High Performance Fortran). OMP. Programación en el modelo de memoria local. Comunicación y granularidad. Programas host-node. Programación en paralelo en el paradigma de intercambios de mensajes: PVM (Parallel Virtual Machine) y MPI (Message-Passing Interface), paquetes con "fuentes abiertos" (open source). Eficiencia de programas paralelos. Speedup y eficiencia. Ley de Amdahl. Factores que influyen en la eficiencia. Sobrecargas por comunicación y coordinación. Fracciones no paralelizables. Redundancia. Desbalanceo. Algoritmos paralelos para álgebra lineal. Bibliotecas de procedimientos para álgebra lineal: caso del ScaLAPACK (open source). Programas paralelos para operaciones matriciales. Otras herramientas (PETSc). Métodos directos de resolución de sistemas de ecuaciones algebraicas lineales. Programación paralela de métodos directos para sistemas de ecuaciones lineales. Sistemas triangulares. Factorización LU. Métodos iterativos de resolución de sistemas de ecuaciones algebraicas lineales. Métodos iterativos para sistemas de ecuaciones lineales. Método de gradiente conjugado preconditionado. Programación paralela. Métodos de descomposición del dominio. Métodos de descomposición del dominio. Complemento de Schur. Aplicación a la resolución de problemas de elementos finitos. Método del complemento de Schur dual. Estrategias para descomponer el dominio en subdominios. Criterios de optimización. Paralelización en programas de elementos finitos. Estructura de un programa de elementos finitos. Tareas desacopladas, débilmente acopladas

y fuertemente acopladas. Paralelización de programas secuenciales. Escritura de programas paralelos. Programa PETSc-FEM.

Conocimientos previos requeridos

Se requieren conocimientos previos sobre cálculo numérico y programación con algún lenguaje científico (FORTRAN, C,C++). Algunas de las aplicaciones están pensadas para estudiar problemas de mecánica de sólidos o fluidos mediante el método de los elementos finitos, por lo que sería deseable además algún conocimiento sobre este método.

Carga horaria: Teoría: 60 horas. Coloquio y/o práctica en aula o laboratorio: 15 horas. Total: 75 horas.

Duración: 15 semanas.

Formas de evaluación: Trabajos prácticos, 2 exámenes parciales y examen final.

Bibliografía básica

Buyya R. (ed.), "High Performance Cluster Computing: Architectures and Systems", Vol.1 y 2, Prentice Hall PTR, NJ, USA, 1999.

<http://www.csse.monash.edu.au/~rajkumar/cluster/index.html>

Deghilage G., "Architectures et Programmation Parallèles", Addison-Wesley, Paris, 1992.

⊃ J.J. Dongarra, Performance of various computers using standard linear equation solvers. <http://www.netlib.org/benchmark/performance.ps>

⊃ Dongarra J., Duff I., Sorensen D. and Van der Vorst H., "Numerical Linear Algebra for High-Performance Computers", SIAM Publication, Philadelphia PA, 1998.

Farhat Ch., "An Introduction to Parallel Scientific Computations", University of Liège, Belgium, 1991.

⊃ Foster I., "Designing and Building Parallel Programs", <http://www.mcs.anl.gov/dbpp/>, on line.

Fox G., Williams R., and Messina P., "Parallel Computing Works", <http://www.npac.syr.edu/copywrite/pcw/>, on line, 1994.

Geist et al., "PVM 3 User's Guide and Reference Manual", Rep. ORNL/TM-12187, Oak Ridge National Laboratory, Oak Ridge, Tennessee USA, 1993. En <http://www.netlib.org/pvm3/>

Golub G. and Van Loan Ch.F., "Matrix Computations", John Hopkins Univ. Press, 1991.

Gropp W., Lusk E. and Skjellum A., "Using MPI: Portable Parallel Programming with the Message- Passing Interface", The MIT Press, Cambridge, 1999.

Koelbel C.H., Loveman D.B., Schreiber R.S., Steele G.L., Zosel M.E., "The High Performance Fortran Handbook", The MIT Press, 1997.

OpenMP Fortran (version 2.0, November 2000) y OpenMP C/C++ (version 2.0, November 2001) especificaciones, <http://www.openmp.org/specs/>

Papadrakakis M. (ed.), "Parallel Solution Methods in Computational Mechanics", J.Wiley and Sons, Chichester, 1997.

⊃ Sterling T.L., Salmon J., Becker D.J. and Savarese D.F., "How to build a Beowulf", The MIT Press, Cambridge, 1999.

Storti M. A. et al, "PETSc-FEM: A General Purpose, Parallel, Multi-Physics FEM Program", <http://minerva.arcrude.edu.ar/PETSc-FEM/>, on line.

=Succi S. and Papetti F., "An introduction to parallel computational fluid dynamics", Nova Science Publ., New York, 1996.
Adaptor 12.0 Parallel Fortran Compilation System.

MÉTODOS NUMERICOS EN FENOMENOS DE TRANSPORTE

Objetivos

Introducción de los métodos numéricos para resolver diferentes problemas de transporte de masa, cantidad de movimiento y energía. El enfoque será centrado hacia temas de índole numérico donde la necesidad de una estabilización espacial amerita tal aproximación. La primera parte del curso será aplicada al caso de transporte de escalares mientras que la parte final se orientará hacia campos vectoriales como el de la fluidodinámica. La aproximación numérica a presentar será la de volúmenes finitos y elementos finitos.

Programa sintético

Introducción a los modelos matemáticos de las ecuaciones de transporte. Discretización por volúmenes finitos. Discretización por elementos finitos. Aplicación a la resolución de problemas escalares lineales, transporte de energía o de contaminantes. Aplicación a la resolución de problemas escalares no lineales, modelo de Burgers como una primera aproximación a las ecuaciones de la mecánica de fluidos. Aplicación a la resolución de problemas vectoriales lineales, ecuaciones de la acústica, modelo de aguas poco profundas y sistemas multicomponentes. Aplicación a la resolución de problemas vectoriales no lineales. Ecuaciones de dinámica de gases , Euler, Navier-Stokes.

Conocimientos previos requeridos

Cálculo numérico y mecánica de fluidos. Recomendable algún curso sobre fenómenos de transporte, transferencia de masa o de energía.

Carga horaria: Teoría. 30 h. Coloquio y/o práctica en aula o laboratorio: 30 horas. Total: 60 horas.

Duración: 15 semanas.

Formas de evaluación: 2 exámenes parciales y examen final.

Bibliografía básica

- Donea J. y Huerta A., "Finite Element Methods for Flow Problems", John Wiley & Sons, 2003.
- Hirsch C., "Numerical Computation of Internal and External Flows", John Wiley & Sons, 2007.
- Leveque R., "Finite Volume Methods for hyperbolic Problems". Cambridge University Press, 2002

MÉTODOS ITERATIVOS PARA LA SOLUCIÓN DE GRANDES SISTEMAS DE ECUACIONES LINEALES Y NO LINEALES

Objetivos

El objetivo de este curso es brindar al estudiante de posgrado una base matemática e informática en lo que respecta a la solución de grandes sistemas de ecuaciones lineales y no lineales que provienen, generalmente, de la discretización de ecuaciones diferenciales en derivadas parciales. Se presentarán las bases matemáticas necesarias para el estudio de convergencia, estabilidad y precisión de métodos iterativos que son de uso extensivo por la comunidad científica para la solución de grandes sistemas de ecuaciones; como así también, se darán detalles de la implementación de estos algoritmos en plataformas de cálculo secuenciales y paralelas con el posterior análisis de su desempeño. Se abordarán temas de constante estudio y desarrollo como lo es el método de Descomposición de Dominios y su preconditionamiento. La idea del curso es también la de usar y evaluar la performance de los métodos iterativos dentro del contexto de la Mecánica Computacional.

Programa sintético

Conceptos Básicos y Métodos Iterativos Estacionarios. Revisión y notación. El lema Banach e inversas aproximadas. El radio espectral. Métodos iterativos estacionarios clásicos. Ejercicios. Método Iterativo de los Gradientes Conjugados (CG). Métodos de Krylov y la propiedad de minimización. Consecuencias de la propiedad de minimización. Criterios de terminación de la iteración. Implementación. Precondicionamiento. Métodos CGNR y CGNE. Ejemplos del método de CG con preconditionamiento. Ejercicios. Iteración de GMRES. La propiedad de minimización y sus consecuencias. Criterios de terminación de la iteración. Precondicionamiento. Implementación de GMRES: ideas básicas. Implementación en una base ortogonal. Colapso de GMRES (Breakdown). El algoritmo de Gram-Schmidt modificado. Una Implementación eficiente. Estrategias de reortogonalización. Restart. Ejemplos para el método de GMRES. Ejercicios. Conceptos Básicos en Iteración de Punto Fijo. Tipos de convergencia. Iteración de punto fijo. Hipótesis estándares. Método de Newton. Convergencia local del método de Newton. Criterios de terminación de la iteración. Implementación del método de Newton. Errores en la función y en su derivada. El Método de Descomposición de dominios. Condicionamiento del problema de interfase. Análisis de Fourier. Problema de Poisson. Problema de Advección-Difusión. Resolución en Plataformas Paralelas de Problemas Lineales y No Lineales usando PETSc. Introducción a la Librería PETSc. Resolución de la Ecuación de Laplace. Resolución de la Ecuación de Advección-Difusión. Resolución de Problemas No Lineales.

Conocimientos previos requeridos

Nociones básicas de Matemática y Álgebra Lineal. Conocimiento de lenguajes de programación como C/C++ y/o Matlab/Octave.

Carga horaria: Teoría. 60 h. Coloquio y/o práctica en aula o laboratorio: 15 horas. Total: 75 horas.

Duración: 15 semanas.

Formas de evaluación: 3 exámenes parciales y examen final.

Bibliografía básica

Golub G., Van Loan Ch. F., "Matrix Computations", John Hopkins Univ. Press, 1991.

Kelley C. T., "Iterative Methods for Linear and Nonlinear Equations". Frontiers in Applied Mathematics, vol. 16. SIAM: Philadelphia, PA, 1995.

Papadrakakis M. (ed.), "Parallel Solution Methods in Computational Mechanics", John Wiley & Sons, Chichester, 1997.

Storti M.A., "Métodos Iterativos para la Solución de Problemas Lineales y No-Lineales". Universidad Nacional del Litoral - CONICET. 2002.

Van der Borst H., "Iterative Methods for Large Linear Systems". Mathematical Institute, Utrech University. The Netherlands, 2000.

ALGORITMOS Y ESTRUCTURAS DE DATOS

Objetivos

Que el alumno conozca las estructuras de datos fundamentales y domine los algoritmos para manipularlas en forma eficiente. Que aprenda a elegir correctamente las estructuras de datos usuales y la implementación para obtener el algoritmo más eficiente para un problema dado. Que sepa diseñar e implementar nuevas estructuras. Que conozca los tiempos de ejecución de los algoritmos fundamentales y determinar los tiempos de ejecución de nuevos algoritmos. Que conozca conceptos de diseño de nuevos algoritmos.

Programa sintético

Diseño y análisis de algoritmo. Conceptos básicos de algoritmos. Tipos abstractos de datos. Conteo de operaciones para el cálculo del tiempo de ejecución. Tipos de datos abstractos fundamentales. El TAD Lista. El TAD pila. El TAD cola. El TAD correspondencia. Árboles. Operaciones con árboles. Interfase básica para árboles. El tipo iterator. Tiempos de ejecución Arboles binarios. Arboles de Huffman. Conjuntos. Implementación por vectores de bits. Implementación con listas. Interfase avanzada para conjuntos. El diccionario. Conjuntos con árboles binarios de búsqueda. Ordenamiento. Métodos de ordenamiento lentos. Ordenamiento indirecto. El método de ordenamiento rápido, quick-sort. Ordenamiento por montículos. Ordenamiento por fusión. Comparación de algunas implementaciones de algoritmos de ordenamiento. Técnicas de Análisis de Algoritmos. Técnicas de Diseño de Algoritmos. Tópicos avanzados.

Conocimientos previos requeridos

Buen conocimiento previo de programación en C/C++. Matemática básica.

Carga horaria: Teoría. 25 h. Coloquio y/o práctica en aula o laboratorio: 15 horas. Total: 40 horas.

Duración: 15 semanas.

Formas de evaluación: Trabajos prácticos, 3 exámenes parciales y examen final.

Bibliografía básica

Aho A. V., Hopcroft J. E. y Ullman J. D., "Estructura de Datos y Algoritmos. Apuntes de la cátedra", <http://www.cimec.org.ar/aed>, , Ed. Addison-Wesley, 1988.

Tenenbaum A. y Augenstein M. J., "Estructura de Datos en Pascal", Ed. Prentice-Hall, 1983.

Weiss M. A., "Estructuras de Datos y Algoritmos", Ed. Addison Wesley, 1995.

Wirth N., "Algoritmos y Estructura de Datos", Ed. Prentice Hall, 1987.

SIMULACION DE MOTORES DE COMBUSTION INTERNA

Objetivos

Repasar fundamentos teóricos. Presentar los modelos más utilizados en la descripción del funcionamiento de un motor de combustión interna desde el punto de vista mecánico, termodinámico y fluidodinámico. Enseñar a desarrollar software para simular motores de combustión interna usando modelos termodinámicos y/o modelos multidimensionales. Analizar algunos componentes de trenes de válvulas y tren inferior. Sistemas de inyección. Contaminación ambiental y dispositivos para minimizar su impacto.

Programa sintético

Modelos termodinámicos o cero-dimensionales. Introducción a los modelos físicos en motores de combustión interna de 4 tiempos encendidos por chispa. Modelo matemático asociado a la dinámica de gases y a la termodinámica del fluido activo en un motor de combustión interna. Esquemas numéricos usados para resolver el modelo matemático. El problema de las condiciones de contorno. Dinámica de sistemas de admisión y escape en un motor monocilíndrico. Simulación cero-dimensional de un motor pluricilíndrico. Modelos de combustión multizonales en cámaras de combustión turbulentas. Aspectos complementarios relacionados con motores diesel y motores 2 tiempos. Modelos multidimensionales. Introducción a la modelización multidimensional en flujos compresibles e incompresibles en dominios fijos y variables. Una breve descripción de los principales modelos físicos usados por KIVA: a. spray de combustible, b. encendido por chispa, c. modelos de combustión, d. modelos de turbulencia, e. modelos para las reacciones químicas de oxidación del combustible. Estructura del código KIVA III y su uso para simular flujos compresibles reactivos en cámaras de combustión.

Conocimientos previos requeridos

Termodinámica, mecánica de fluidos, máquinas térmicas y cálculo numérico (deseable).

Carga horaria: Teoría. 30 h. Coloquio y/o práctica en aula o laboratorio: 30 horas. Total: 60 horas.

Duración: 15 semanas.

Formas de evaluación: 1 examen parcial y examen final.

Bibliografía básica

- Amsden A., O'Rourke P. J, and Butler C. D., "KIVA II: A computer program for chemically reactive flows with sprays". Los Alamos National Laboratory, 1989.
- Amsden A., O'Rourke P. J, and Butler C. D., "KIVA III: A KIVA program with Block-Structured mesh for complex geometries". Los Alamos National Laboratory, 1993.
- Allesandri M., "Simulazione della fase di recambio della carica in un motore benzina e iniezione diretta". Tesi di Laurea di la Universita Degli Studi di

Roma Tor Vergata.

Bella G., "Notas del curso de posgrado de la Universita Degli Studi di Roma Tor Vergata.

Benson R., "The Thermodynamics and Gas Dynamics of Internal Combustion Engines", Vol I, Clarendon Press, Oxford, 1982.

Heywood J., "Internal combustion engines fundamentals". McGraw-Hill, 1989

Ramos J., "Internal combustion engine modeling". Hemisphere Publishing Corporation, 1989.

GEOMETRÍA COMPUTACIONAL

Objetivos

Se pretende dotar al alumno de las competencias necesarias para analizar y diseñar algoritmos eficientes para la resolución de problemas que pueden plantearse en términos geométricos. La presentación se hará a través de una serie de algoritmos ejemplares, seleccionados para entender los conceptos generales. Se estudiará la eficiencia y robustez de los algoritmos y los métodos para abordar los casos patológicos. El objetivo primario es que el alumno aprenda a razonar con la lógica particular de éste tipo de algoritmos.

Programa sintético

Introducción: Definición y alcances de la Geometría Computacional. Conceptos previos: Series numéricas. Elementos de la Topología: Abierto, Espacio Topológico, Entorno, Frontera, Conexidad, Compacidad, Homeomorfismo, Manifolds. Espacios y transformaciones: Vectorial, Afín, Proyectivo, Métricas. Coordenadas en símlices y funciones de forma. Análisis Asintótico y Complejidad algorítmica: Cotas O , o , Ω , ω y Θ ; Análisis de recursiones y Master Theorem. Envoltorio Convexo en el plano: Clasificación de polígonos: Simple/Autointerceptado Convexo/Star-shaped/Visible desde el exterior. Monótono respecto a una línea. Definiciones equivalentes de Envoltorio Convexo. Algoritmo $O(n^3)$ trivial. Algoritmos $O(n \log(n))$: Incremental: Graham Scan. Divide and Conquer Quick Hull Algoritmos output-dependent: $O(nh)$: Gift Wrapping y Jarvis March $O(n \log(h))$: Algoritmo de Chan Búsqueda en $O(\log(n))$ de tangentes y Bounding-Box de polígonos convexos. Intersecciones de segmentos en el plano: Algoritmo de barrido plano y estructuras de datos asociadas: Cola de Eventos. Árboles Binarios Auto-Balanceados. Lista Doblemente Enlazada de aristas. Operaciones Booleanas entre grafos (mallas) y regiones planas (GIS). Triangulación de Polígonos Simples. Problema del guardián de la galería de arte y triangulaciones. Art Gallery Theorem. Partición de un polígono en piezas monótonas: Algoritmo de barrido. Triangulación de polígonos monótonos en tiempo lineal. Triangulación de polígonos con fronteras interiores, regiones planas con líneas sueltas y nubes de puntos aislados. Programación Lineal. Desmoldabilidad de poliedros y chapas estampadas. Intersección de semiplanos: Algoritmo divide and conquer. Programación lineal: Algoritmo incremental. Justificación y backwards analysis para los Algoritmos Aleatorios: Expected Time. Unbounded Linear Programs Programación lineal en muchas dimensiones. Problemas parecidos o LP-type problems. Facility planning, Euclidean 1-center o envoltorio esférico. Normales y mínimo casquete envolvente en la esfera unitaria. Ubicación de puntos: Bases de datos y el problema 1D. Búsqueda ortogonal. Trees y Tries: kD Tree. Octree. BSP tree. Búsqueda en triangulaciones por funciones de forma. Dualidades: Grafos duales. Incidencia. Dualidad punto/línea: Dualidad Algebraica. Inversión circular - Polaridad. Dualidad

Proyectiva y polaridad cónica. Diagrama de Voronoï y Triangulación Delaunay. Definiciones y dualidad, propiedades. Algoritmo de Fortune: lifting map y coastline. Triangulación incremental aleatoria. Generación de puntos y mallas. Algoritmos en 3D. Algoritmos en muchas dimensiones.

Conocimientos previos requeridos

Conocimientos básicos de programación y de geometría. Preferentemente: conocimientos de algoritmos y estructuras de datos, computación gráfica y programación en C.

Carga horaria: Teoría. 45 horas. Coloquio y/o práctica en aula o laboratorio: 45 horas. Total: 90 horas.

Duración: 15 semanas.

Formas de evaluación: Trabajos prácticos, 1 examen parcial y examen final.

Bibliografía básica

Boissonnat J. D. and Yvinec M., "Algorithmic Geometry", Cambridge University Press, 1997.

De Berg M., van Kreveld M., Overmars M., Schwarzkopf O., "Computational Geometry: Algorithms and Applications". Springer-Verlag, 2nd ed., 2000.

O'Rourke J. "Computational Geometry in C". Cambridge University Press, 2nd ed., 1998.

Preparata F. P. and Shamos M. I., "Computational Geometry: An Introduction", Springer-Verlag, 1985.

Lecture notes:

Erickson J., "CS 473: Algorithms". University of Illinois at Urbana-Champaign, Department of Computer Science, 2007.

Mount D., "CMSC 754: Computational Geometry". University of Maryland, Department of Computer Science, 2007

COMPUTACIÓN DE ALTO RENDIMIENTO EN MECÁNICA COMPUTACIONAL. MPI, PETSC Y OPEN MP

Objetivos

El objetivo del curso es desarrollar las aptitudes del alumno en cuanto a la programación en C++ con los estándares/librerías MPI, PETSc y OpenMP, los cuales son actualmente una combinación efectiva para la resolución de grandes problemas de mecánica computacional en cluster de procesadores multi-core. La orientación del curso no es hacia el aprendizaje general de la computación en paralelo sino al uso intensivo de estas tecnologías.

Programa sintético

Conceptos básicos de MPI. Es MPI pequeño o grande? Uso de MPI en programas simples. Ejemplos en Fortran y C. Tomando tiempos de programas. Un ejemplo con self-scheduling. I/O en MPI. Estrategia master/slave en SPMD. Formato de llamadas, códigos de error. Funciones básicas. Comunicación punto a punto. Envelope de un mensaje, condiciones de recepción. Dead-lock. La función send-recv. Comunicaciones no-bloqueantes. Ejemplos: Cálculo del ancho de banda de una red. Cálculo del ancho de banda de disección de un cluster. Implementación de broadcast lineal y en árbol con comunicación punto a punto. Comunicación colectivas. Broadcast. Reducción global. Operaciones asociativas. Definición de nuevas operaciones asociativas. Allreduce. Tiempos de comunicación y sincronización. Escalabilidad. Ejemplo: Cálculo de PI por integración. Llamadas colectivas. MPI en ambientes Unix. MPICH. Utilitarios. Ejemplo: Prime Number Theorem. Estrategia compute-on-demand. MPE logging. Escalabilidad. Usod e jumpshot. Rendimiento en clusters heterogéneos. Paralelismo trivial. Ejemplo: El Problema del Agente Viajero(TSP). Adaptación del algoritmo dinámico. Ejemplo: Cálculo de PI por Montecarlo. Uso de communicators. Ejemplo: producto de matrices en paralelo. Algoritmos estático y dinámico. Escalabilidad. Ejemplo: El Juego de la Vida de Conway. Automatas celulares. Algoritmo estático y dinámico. Comunicación encadenada. Escalabilidad. Comunicación no-bloqueante. Ejemplo: El problema de Poisson. Topologías virtuales. Escalabilidad. Operaciones colectivas avanzadas de MPI. Scatter, gather y all-gather. Versiones vectorizadas. Ejemplo: la función print-par (gather and print). Ejemplo: La función rescatter. Definiendo tipos de datos derivados. Ejemplo: rotar columnas de matrices. Ejemplo: El método iterativo de Richardson. PETSc. La librería PETSC de álgebra lineal en paralelo. Objetos PETSc. Estructura de la librería PETSc. Usando PETSc. Escribiendo programas que usan PETSc. Ejemplo simple. Ec. de Laplace 1D. Headers. Bases de datos/Opciones. Vectores y datos distribuidos. Creando vectores. Operaciones básicas sobre vectores. Indiciación. Operaciones de scatter y gather. Matrices. Matrices sparse. Matrices densas. Operaciones básicas sobre matrices. Operaciones 'matrix-free'. SLES: Solvers lineales . Resolviendo secuencias de sistemas lineales.. Familia de métodos de Krylov . Precondicionamiento. Tests de convergencia. Monitoreo de la convergencia. Espectro del operador. OpenMP. Conceptos básicos de OpenMP. Memoria compartida. Datos privados y públicos. Race conditions. El OpenMP estándar. Threads. Regiones paralelas. Pragmas. Cláusulas. Interacción con OpenMP: directivas, variables de entorno, entorno en tiempo de ejecución. Compilación condicional. Lazos paralelos. Sincronización. Cláusulas private, shared,

first/last private, reduction. Directivas de trabajo compartido: parallel. OpenMP. Balance de carga: la cláusula `schedule`. Secciones. Directivas huérfanas de trabajo compartido. Sincronización con barreras. Regiones críticas. Directivas single and master. Otras directivas de sincronización. Ejemplo: Productos de muchas matrices. Ejemplo: Producto matriz vector. OpenMP. Variables de entorno. Funciones de la librería en tiempo de ejecución. Rutinas utilitarias de información y ajuste de parámetros. Rutinas para el manejo de semáforos. Semáforos anidados. Ejemplo: simulando scheduling dinámico con semáforos.

Conocimientos previos requeridos

Programación en C++. Álgebra lineal.

Carga horaria: Teoría. 60 horas. Coloquio y/o práctica en aula o laboratorio: 15 horas. Total: 75 horas.

Duración: 15 semanas.

Formas de evaluación: Trabajos prácticos, 2 exámenes parciales y examen final.

Bibliografía básica

- Balay S., Buschelman K., Eijkhout V., Gropp W., Kaushik D., Knepley M., Curfman McInnes L., Smith B., Zhang H., "PETSc, Portable, Extensible Toolkit for Scientific Computation", User manual, ([download](#))
- Gropp W., Lusk E. and Skjellum A., "Using MPI: Portable Parallel Programming with the Message-Passing Interface", Cambridge University Press, 2003.
- Karniadakis G., Kirby R.M., "Parallel Scientific Computing in C++ and MPI: A Seamless Approach to Parallel Algorithms and their Implementation", Cambridge University Press, 2003.
- Ortega J., "Introduction to Parallel and Vector Solution of Linear Systems", Plenum Press, New York, 1988.
- Papadrakakis M., "Solving Large-Scale Problems in Mechanics", J.Wiley and Sons, 1993.
- Storti M. A., Apuntes del curso ([download](#)).
- Papadrakakis M. (ed.), "Parallel Solution Methods in Computational Mechanics", J.Wiley and Sons, Chichester, 1997.
- Sterling T.L., Salmon J., Becker D.J. and Savarese D.F., "How to build a Beowulf", The MIT Press, Cambridge, 1999.
- Storti M. A., "PETSc-FEM: A General Purpose, Parallel, Multi-Physics FEM Program", <http://www.cimec.org.ar/petscfem>, online.
- Succi S. and Papetti F., "An introduction to parallel computational fluid dynamics", Nova Science Publ., New York, 1996.
- Van der Pas Ruud., Scalable Systems Group, Sun Microsystems, "An Introduction Into OpenMP, IWOMP 2005", University of Oregon, Eugene, Oregon, USA, 2005.

METODOS NUMERICOS EN MECANICA DE FLUIDOS

Objetivos

Brindar las herramientas mínimas necesarias para poder usar y desarrollar software relacionado con la resolución de una gran variedad de modelos matemáticos muy usados en la Mecánica de Fluidos, en especial en situaciones donde existe flujo de fluidos incompresibles y compresibles, laminares y turbulentos, viscosos e invíscidos, estacionarios y transitorios en dominios unidimensionales, bidimensionales y tridimensionales. Dotar al alumno de conocimientos para poder analizar los modelos matemáticos mas usados en la simulación computacional de la mecánica de fluidos. Enseñar a usar software para resolver problemas académicos con especial énfasis en la introducción de parámetros, condiciones de contorno y condiciones iniciales.

Programa sintético

Modelos numéricos en sistemas advectivos difusivos generales. El caso de flujo compresible y shallow-water. Modelos numéricos en flujo incompresible. Modelado numérico de la turbulencia.

Conocimientos previos requeridos

Mecánica de fluidos. Cálculo numérico.

Carga horaria: Teoría. 30 horas. Coloquio y/o práctica en aula o laboratorio: 30 horas. Total: 60 horas.

Duración: 15 semanas.

Formas de evaluación: 2 exámenes parciales y examen final.

Bibliografía básica

Ferziger J. and Peric M., "Computational Methods for fluid dynamics". Ed. Springer 1996.

Hirsch C., "Numerical Computation of Internal and External Flows", John Wiley & Sons, 2007.

Hoffmann K. and Chiang S., "Computational Fluid Dynamics Volume I y II", Engineering Education System, 2000.

Laney C. B., "Computational Gas Dynamics", Cambridge University Press, 1998.

Wilcox D.C., "Turbulence Modeling for CFD", DCW Industries Inc., 2004.

INTRODUCCIÓN AL CÁLCULO TENSORIAL Y SUS APLICACIONES EN MECÁNICA DEL CONTINUO

Objetivos

Presentar las herramientas básicas del álgebra y análisis tensorial para su aplicación en problemas de la mecánica de los medios continuos y en otras ramas de la física y la química donde se utilicen conceptos tensoriales.

Programa sintético

Álgebra de vectores y tensores cartesianos. Cálculo diferencial e integral con vectores y tensores cartesianos. Una introducción al álgebra y el cálculo con tensores generales. Introducción a la mecánica del continuo. Sus aplicaciones en mecánica de sólidos y de fluidos.

Conocimientos previos requeridos

Álgebra lineal. Cálculo diferencial e integral.

Carga horaria: Teoría: 30 horas. Coloquio y/o práctica en aula o laboratorio: 30 horas. Total: 60 horas.

Duración: 15 semanas.

Formas de evaluación: 1 examen parcial y examen final.

Bibliografía básica

- Aris R., "Vectors, Tensors, and the basic equations of fluid mechanics", Courier Dover Publications, 1989.
- Brannon R., "Functional and structured Tensor Analysis for Engineers". <http://www.mech.utah.edu/~brannon/public/Tensors.pdf>
- Chorlton F., "Vector and tensor methods", E. Horwood Ltd., 1976.
- Etter D., "Engineering problems solving with Matlab", Prentice Hall, 1996.
- Heinbockel J., "Introduction to tensor calculus and continuum mechanics", National Library of Canada, 2001.
- Malvern L., "Introduction to the mechanics of a continuous medium", Michigan State University, Dept. of Metallurgy, Mechanics and Materials Science, 1962.
- Reddy J., "An introduction to the Finite Element Method", McGraw-Hill, 2005.
- Soto Prieto M. y Córdoba J., "Álgebra lineal con Matlab y Maple", Prentice Hall, 1995.
- Ogden R., "Non-linear elastic deformations", Dover Publications, 1997.

PROGRAMACIÓN / COMPUTACIÓN CIENTÍFICA CON FORTRAN 95

Objetivos

Fortran 95 es un lenguaje sumamente eficiente para la modelación matemática, y a pesar de no estar totalmente orientado a objetos, incluye y soporta una cantidad de atributos que hacen factible abarcar dicho paradigma de la programación moderna (que ya forma parte del nuevo estándar F2003). El curso introduce al alumno al uso de conceptos básicos de la programación científica utilizando la nueva sintaxis del F95. En la 1ra parte se estudian ejemplos básicos destinados a presentar la funcionalidad esencial del lenguaje. En la 2da parte se resuelven problemas prácticos del continuo y del análisis numérico para que el alumno incorpore el esquema de resolución de un problema técnico específico mediante el planteo y diseño de un algoritmo, su programación acorde a la sintaxis requerida por el F95, la compilación con rutinas científicas y la generación e interpretación de los resultados. En esta última parte, se desarrolla el concepto de error numérico, estabilidad, convergencia, etc. Se provee un compilador para la realización de los trabajos prácticos en gabinete de informática, y los exámenes parcial y final están concebidos para que el alumno demuestre, a tiempo real, su habilidad en el uso y manejo del lenguaje.

Programa sintético

Aspectos de Programación / Fortran 95 Esencial. Introducción: Acerca del curso. Bibliografía. Un poco de historia. Rasgos salientes de F95. Compiladores F95 disponibles (gratuitos y comerciales). Elementos de F95: Tipos de datos intrínsecos y derivados (definidos por el usuario), Cadenas de caracteres, Arreglos. Estructura del programa principal, Unidades de programas y procedimientos, Subrutinas y funciones, Interfaces (explícitas e implícitas), Uso de módulos en reemplazo de interfaces. El concepto de "module" y "use". Funciones intrínsecas size, shape, maxloc, maxval, minloc, minval, count, sum, etc. Sobrecarga de operadores (polimorfismos). Sentencias de control (if, case, do, do while, cycle, etc). Arreglos de orden cero y superiores, constructores de arreglos. Asignación dinámica de arreglos allocate & deallocate). Punteros. Operaciones con índices en F95. Sintaxis matricial con un estilo cuasi Octave/Matlab. Construcción Where ... Else. Diferencias entre las instrucciones Do y Forall. Computación Científica. Aproximación e interpolación de funciones, matrices en diferencias, clasificación de ecuaciones diferenciales en derivadas parciales, problemas de valores iniciales y de contorno, solución por diferencias finitas, problemas de autovalores y autovectores, solución de la ecuación de difusión 1D por elementos finitos, advección-difusión estacionaria 1D.

Conocimientos previos requeridos

Familiaridad con PCs, y con Análisis Numérico (no excluyente).

Carga horaria: Teoría: 45 horas. Coloquio y/o práctica en aula o laboratorio: 45 horas. Total: 90 horas.

Duración: 15 semanas.

Formas de evaluación: Trabajos prácticos, 1 examen parcial y examen final.

Bibliografía básica

Decyk V, "Scientific Computing with Fortran 95", PSTI Research Lecture Series, UCLA, USA, 2002.

D'Elía J., "Fortran 95", notas de clase, 2005.

Metcalf M., Reid J., Cohen M., "Fortran 95/2003 explained", Oxford University Press, 2004.

Metcalf M., Reid J., "Fortran 90/95 explained", Oxford Science Publications, 1996.

Redwine C, "Upgrading to Fortran 90", Springer, 1995.

Trefethen L., Spectral methods in MATLAB, SIAM, 2001.