

## 1. Lógica computacional

En este documento se recogen algunos contenidos que se pueden incluir en la línea “Lógica computacional” (LC) del Máster Universitario en Matemática Avanzada.

Se presenta desde tres puntos de vista:

1. Contenidos de LC según la clasificación de la AMS.
2. Contenidos de LC según “Handbook of practical logic and automated reasoning”.
3. Contenidos de LC según casos de estudio de teorías formalizadas.

### 1.1. Contenidos de LC según la clasificación de la AMS

- Algoritmos en lógica proposicional clásica (03B05, 03B35).
- Algoritmos en lógica de primer orden (03B10, 03B35).
- Lógica de orden superior y teoría de tipos (03B15).
- Decidibilidad de teorías (03B25).
- Automatización de pruebas y operaciones lógicas (03B35).
- Lambda cálculo (03B40).
- Lógicas del conocimiento y creencias (03B42).
- Lógica temporal (03B44).
- Lógicas modales y descriptivas (03B45).
- Lógicas multivalentes (03B50).
- Lógica en las ciencias de la computación (03B70).
- Eliminación de cuantificadores.
- Números ordinales y pruebas de terminación (03E10)
- Programación lógica (68N17)
- Programación funcional y lambda cálculo (68N14)
- Aspectos matemáticos de la programación (especificación, verificación, etc.) (68N30).
- Modelos de computación (68Q05).
- Análisis de algoritmos (68Q25).
- Lógica ecuacional y sistemas de reescritura (68Q42).
- Especificación y verificación (68Q60)
- Especificaciones algebraicas (68Q65)
- Demostración automática de teoremas (68T15)
- Lógica en inteligencia artificial (68T27)
- Representación del conocimiento (68T30)
- Tecnología de agentes (68T42)

## 1.2. Contenidos de LC según “Handbook of practical logic and automated reasoning”

En el libro Handbook of practical logic and automated reasoning<sup>1</sup> se presenta la implementación de los algoritmos fundamentales de la lógica y de la automatización del razonamiento.

Este libro es la base de distintos cursos por ejemplo Lógica en Haskell<sup>2</sup> o Topics in Automated Deduction<sup>3</sup>

Su contenido es el siguiente:

1. Algoritmos de la lógica proposicional:
  - 1.1. Sintaxis de la lógica proposicional.
  - 1.2. Semántica de la lógica proposicional.
  - 1.3. Validez y satisfacibilidad.
  - 1.4. Leyes de De Morgan, adecuación y dualidad.
  - 1.5. Forma normal negativa.
  - 1.6. Forma normal conjuntiva.
  - 1.7. Aplicaciones de la lógica proposicional.
  - 1.8. FNC con definiciones.
  - 1.9. El procedimiento de Davis-Putnam
  - 1.10. El método de Stalmarck.
  - 1.11. Diagramas de decisión binarios.
  - 1.12. Compacidad.
2. Algoritmos de la lógica de primer orden:
  - 2.1. Sintaxis de la lógica de primer orden.
  - 2.2. Semántica de la lógica de primer orden.
  - 2.3. Operaciones sintácticas.
  - 2.4. Forma normal prenexa.
  - 2.5. Forma de Skolem.
  - 2.6. Modelos canónicos.
  - 2.7. Mecanización del teorema de Herbrand.
  - 2.8. Unificación.
  - 2.9. Tableros semánticos.
  - 2.10. Resolución.
  - 2.11. Subsunción.
  - 2.12. Refinamientos de resolución.
  - 2.13. Cláusulas de Horn y Prolog.
  - 2.14. Eliminación de modelos.
3. Algoritmos sobre la lógica con igualdad:
  - 3.1. Axiomas de la igualdad.
  - 3.2. Categoricidad y equivalencia elemental.
  - 3.3. Lógica ecuacional y teoremas de completitud.
  - 3.4. Clausura por congruencias.
  - 3.5. Reescritura.
  - 3.6. Órdenes de terminación.
  - 3.7. Método de completación de Knuth-Bendix
  - 3.8. Eliminación de la igualdad.

---

<sup>1</sup><http://bit.ly/HIAdr2>

<sup>2</sup><http://bit.ly/17YKjOI>

<sup>3</sup><http://bit.ly/17YKCsQ>

- 3.9. Paramodulación.
4. Problemas decidibles:
  - 4.1. El problema de la decisión.
  - 4.2. El fragmento AE.
  - 4.3. El fragmento monádico.
  - 4.4. Silogismos,
  - 4.5. La propiedad de modelo finito.
  - 4.6. Eliminación de cuantificadores.
  - 4.7. Aritmética de Presburger.
  - 4.8. Los números complejos.
  - 4.9. Los números reales.
  - 4.10. Anillos, ideales y problemas de palabra.
  - 4.11. Bases de Gröbner.
  - 4.12. Demostración automática de teoremas geométricos.
  - 4.13. Combinación de procedimientos de decisión.
5. Demostración interactiva de teoremas:
  - 5.1. Métodos de deducción natural.
  - 5.2. Demostradores interactivos y verificadores de pruebas.
  - 5.3. Sistemas de prueba par la lógica de primer orden.
  - 5.4. Implementación de la lógica de primer orden en LCF.
  - 5.5. Reglas proposicionales derivadas.
  - 5.6. Demostración de tautologías mediante inferencias.
  - 5.7. Reglas de primer orden derivadas.
  - 5.8. Demostración de primer orden mediante inferencias.
  - 5.9. Estilos de demostraciones interactivas.
6. Limitaciones de la lógica y su automatización:
  - 6.1. El programa de Hilbert.
  - 6.2. El teorema de Tarski sobre la indefinibilidad de la verdad.
  - 6.3. Incompletitud de sistemas axiomáticos.
  - 6.4. Teorema de incompletitud de Gödel.
  - 6.5. Definibilidad y decidibilidad.
  - 6.6. Teorema de Church.

### **1.3. Contenidos de LC según casos de estudio de teorías formalizadas**

En el libro Isabelle/HOL – Higher-Order Logic<sup>4</sup> se presenta las formalizaciones de las pruebas de distintas teorías. El estudio de algunas de dichas formalizaciones puede servir para ilustrar, como casos de estudio, el contenido del curso.

Algunas de dichas formalizaciones de teorías son las siguientes:

1. HOL: teoría básica de orden superior.
2. Teorías de conjuntos ordenados.
3. Teoría de grupos.
4. Teoría de retículos.
5. Teoría de conjuntos.
6. Teoría de retículos completos.
7. Teorema de Knaster-Tarski del punto fijo y definiciones inductivas.

---

<sup>4</sup><http://bit.ly/19cTM0W>

8. Teoría conjuntista de funciones.
9. Teoría de anillos.
10. Teoría de cuerpos.
11. Teoría de los números naturales.
12. Teoría de conjuntos finitos.
13. Teoría de relaciones.
14. Clausuras reflexiva-tranitiva.
15. Relaciones bien fundamentadas. Resolución bien fundamentada.
16. Demostraciones de terminación de funciones.
17. Teoría de los números enteros.
18. Teoría de la divisibilidad.
19. Bases de Gröbner.
20. Procedimiento de decisión para la aritmética de Presburger.
21. Teoría de listas.
22. Cuerpos arquimedianos.
23. Teoría de los números racionales.
24. Desarrollo de los reales mediante sucesiones de Cauchy.
25. Completitud de los reales positivos.
26. Espacios vectoriales y topologías sobre los reales.
27. Filtros y límites.
28. Sucesiones y convergencia.
29. Límite y continuidad.
30. Derivadas y teoremas del valor medio y de Rolle.
31. Series: Sumas finitas y series infinitas.
32. Raíces de números reales.
33. Transcendentes: Series de potencias, funciones trascendentes, etc.
34. Teoría de números complejos.
35. Teoría de logaritmos.
36. Series de MacLaurin y de Taylor.

Otra fuente de casos de estudios es el de las formalizaciones realizadas en otros sistemas. Por ejemplo, en ACL2<sup>5</sup>, Coq<sup>6</sup> o PVS<sup>7</sup>.

---

<sup>5</sup><http://bit.ly/16eZl0Y>

<sup>6</sup><http://bit.ly/HQ5mte>

<sup>7</sup><http://1.usa.gov/HQ5rwU>

## 2 Modelos de Computación Bioinspirada. Complejidad Computacional y modelización

Se agrupan en este epígrafe tres líneas:

### 2.1 Modelos de Computación Bioinspirada

**CONTENIDO** (palabras clave / temáticas relacionadas):

- Teoría de la Computación.
- Computación Natural.
- Modelos de computación bioinspirados no convencionales (computación molecular con ADN, computación con membranas, y otros).
- Modelos de computación bioinspirados "fuzzy".
- Estudio de la potencia de los modelos. Universalidad. Jerarquías.
- Teoría de la Complejidad Computacional en el marco de los modelos de computación bioinspirados (eficiencia).
- Aplicaciones de estos modelos.
- Verificación formal de los programas / sistemas diseñados.
- Software de simulación.
- Vida Artificial.

### 2.2 Teoría de la Complejidad Computacional

**CONTENIDO** (palabras clave / temáticas relacionadas):

- Computabilidad versus Complejidad.
- Modelos convencionales de computación.
- Recursos computacionales: tiempo y espacio. Complejidad en tiempo y en espacio.
- NP Completitud. P vs NP.
- Complejidad computacional vs complejidad descriptiva.

- Computación paralela.
- Máquinas moleculares basadas en ADN y máquinas celulares con membranas.

## 2.3 Modelización Computacional con Máquinas Celulares

**CONTENIDO** (palabras clave / temáticas relacionadas):

- Modelización de procesos complejos de la vida real.
- Computación celular con membranas (Sistemas P).
- Modelos con semántica probabilística / estocástica. Gillespie.
- Protocolos de modelización.
- Dinámica de poblaciones. Biología de Sistemas. Biología computacional.
- Software de simulación.
- Experimentación virtual, ajuste de parámetros y validación experimental de los modelos diseñados.

## 2.4 Áreas relacionadas

Códigos de la "Mathematics Subject Classification 2010" relacionados con las líneas.

03Dxx Computability and recursion theory

03D10 Turing machines and related notions

03D15 Complexity of computation

(including implicit computational complexity)

03D55 Hierarchies

68Qxx Theory of computing

68Q05 Models of computation (Turing machines, etc.)

68Q10 Modes of computation (nondeterministic, parallel, interactive, probabilistic, etc.)

68Q15 Complexity classes

(hierarchies, relations among complexity classes, etc.)

- 68Q17 Computational difficulty of problems (lower bounds, completeness, difficulty of approximation, etc.)
- 68Q60 Specification and verification (program logics, model checking, etc.)
- 68Q85 Models and methods for concurrent and distributed computing (process algebras, bisimulation, transition nets, etc.)

92-XX Biology and other natural sciences

- 92C42 Systems biology, networks
- 92Dxx Genetics and population dynamics

Tópicos de "The 2012 ACM Computing Classification System" relacionados con las líneas

General and reference

- Cross-computing tools and techniques
- Validation
- Verification

Theory of computation

- Models of computation
  - Computability
    - Lambda calculus
    - Turing machines
    - Recursive functions
- Formal languages and automata theory
  - Formalisms
    - Rewrite systems
- Computational complexity and cryptography
  - Complexity classes
  - Problems, reductions and completeness
  - Algebraic complexity theory
  - Proof complexity
  - Interactive proof systems
  - Complexity theory and logic
- Design and analysis of algorithms

Computing methodologies

- Symbolic and algebraic manipulation
  - Symbolic and algebraic algorithms
- Modeling and simulation

- Simulation types and techniques
  - Molecular simulation
  - Agent / discrete models
- Applied computing
  - Life and medical sciences
    - Computational biology
      - Molecular sequence analysis
      - Recognition of genes and regulatory elements
      - Molecular evolution
      - Computational transcriptomics
      - Biological networks
      - Sequencing and genotyping technologies
      - Computational proteomics
      - Molecular structural biology
      - Computational genomics
    - Genomics
      - Computational genomics
    - Systems biology
    - Bioinformatics

## 2.5 Respaldo (común a las tres líneas)

Estas líneas tienen una fuerte relación con el campo de la teoría de la computación, teoría de autómatas y teoría de lenguajes formales. La Computación no convencional en general, y la Computación natural en particular, son líneas de investigación actualmente en auge. Existe un fuerte componente formal en todo lo relativo a demostraciones (por ejemplo, de universalidad), y también en lo relativo al análisis del coste computacional de las computaciones de los dispositivos diseñados. Por la naturaleza no determinista y masivamente paralela de estos modelos, se establecen vínculos con sistemas complejos, computación paralela, procesos concurrentes, etc. Asimismo, la línea 3 tiene relación con otros marcos de modelización (EDOS, multiagent systems, ...) algoritmos de MonteCarlo / Gillespie.

El Dpto CCIA cuenta con personal con formación y experiencia en estas líneas, habida cuenta de su trayectoria docente e investigadora.



## 3 Modelización y Razonamiento sobre Sistemas Complejos

### 3.1 Contenidos

- Modelado matemático-computacional
- Dinámica de Sistemas
- Sistemas MultiAgente
- Teoría de Redes
- Autómatas Celulares
- Fractales
- Sistemas Distribuidos
- Inteligencia Colectiva
- Dinámica Simbólica
- Algoritmos de Aproximación
- Sistemas de Clasificación y Aprendizaje
- Redes Semánticas
- Análisis Formal de Conceptos

Sin lugar a dudas, la comprensión de los sistemas complejos es uno de los grandes retos de la ciencia en el S.XXI, y dentro de ella, la labor de las matemáticas ha de ser la de definir y formalizar las herramientas necesarias para abordar este reto. Además de la importancia puramente teórica que presenta esta línea en relación con muchas de las ramas clásicas de las matemáticas, desarrollar metodologías de análisis formal sobre ella permitirá ampliar el rango de aplicaciones que ya está empezando a mostrar en multitud de otras áreas del conocimiento, y que van desde la mejor comprensión de sistemas físicos y biológicos, hasta su aplicación al modelado de sistemas culturales, económicos y sociales. En este sentido, varios de los grupos de CCIA han demostrado tener amplia capacidad docente e investigadora, tanto a nivel teórico como en aplicaciones.

## 3.2 Descripción del área

Un sistema puede definirse como un "conjunto de elementos en interacción" (Bertalanffy, 1968). Entre éstos, los sistemas complejos (p. ej. organismos pluricelulares, colonias de hormigas, ecosistemas, economías, sociedades...) se caracterizan por mostrar una dinámica que se resiste a ser modelada con las herramientas matemáticas tradicionales. En estos sistemas complejos (Vicsek, 2002; Gilbert, 2004) se pueden descubrir las siguientes características comunes:

- Muestran diversos niveles jerárquicos de agrupación.
- Los componentes de niveles jerárquicos inferiores suelen mostrar un grado de autonomía significativo.
- El comportamiento del sistema surge a partir de la auto-organización de sus componentes, sin que esta organización esté controlada ni dirigida por ningún ente exterior al sistema. Es decir, son sistemas distribuidos sin centralización.

Además, muchos de estos sistemas son también adaptativos, lo que significa que el comportamiento de sus componentes básicos puede evolucionar en el tiempo, dando lugar a una cierta capacidad de respuesta frente a cambios en el entorno por medio de mecanismos de aprendizaje a escala individual y/o mecanismo de selección y reemplazo (lo cual da lugar a un aprendizaje a escala poblacional).

Todas estas características hacen que el proceso de modelado formal de sistemas complejos difiera sustancialmente del de otros sistemas más simples. En particular, su naturaleza descentralizada, la presencia de bucles de causalidad y retroalimentación no lineales, y el hecho de contener varias unidades más o menos autónomas, que pueden interactuar, evolucionar, y adaptar su comportamiento a cambios en el entorno, implica que en la mayoría de los casos es muy difícil (cuando no imposible, al menos actualmente) conseguir un modelo basado exclusivamente en las herramientas matemáticas clásicas que pueda describir el sistema adecuadamente y que, simultáneamente, sea resoluble.

Hasta hace poco, no poder resolver matemáticamente un modelo formal suponía una desventaja importante, puesto que carecíamos de otras herramientas con las que deducir las implicaciones lógicas que de él se derivaban. Actualmente, la computación matemática permite explorar y analizar modelos formales que no pueden ser resueltos explícitamente por medios clásicos. Como muestra Suber (Suber, 2007), un modelo que está implementado por medios computacionales y que puede ejecutarse en un ordenador es

un modelo necesariamente formal y no difiere sustancialmente de un modelo matemático tradicional. Lo que equipara los modelos computacionales a los modelos matemáticos clásicos, mostrando que el único cambio es en el lenguaje en el que se expresan.

Esencialmente, un modelo formal computacional puede verse como un sistema matemático deductivo particular, los resultados obtenidos al ejecutar la simulación computacional sobre el modelo se derivan por deducción (y, por lo tanto, con necesidad lógica) de aplicar las reglas algorítmicas que definen el modelo al conjunto de condiciones iniciales con las que el modelo se ha parametrizado. De esta forma, independientemente de su mayor o menor complejidad interna, una simulación computacional constituye un sistema de deducción válido.

Entre las metodologías que se pueden usar para el modelado computacional podemos destacar por su importancia metodológica y por los buenos resultados que está cosechando la basada en Sistemas Multi-Agentes.

BERTALANFFY , L. (1968): *General System Theory*. New York: George Braziller Publisher.

VICSEK , T. (2002): *Complexity: The bigger picture*. Nature 418(6894), pp. 131-131.

GILBERT , N. (2004): *Agent-based social simulation: Dealing with complexity*. Centre for Research on Social Simulation, University of Surrey.

SUBER , P. (2007): *Formal Systems and Machines: An Isomorphism*. Hand-out for “Logical Systems”. Earlham College.

### **3.3 Respaldo:**

Actualmente existe un grupo (multidepartamental) de profesores dentro de la U. de Sevilla trabajando en este acercamiento a los sistemas complejos, así como proyectos de excelencia y del Ministerio que están dedicados, en gran parte, a la aproximación antes descrita.

## 4 Teoría de modelos y aplicaciones

### 4.1 Contenido

- Lenguajes y estructuras de primer orden.
- Eliminación de cuantificadores, modelo-completitud.
- Estructuras y modelos finitos.
- Ultraproductos.
- Forcing en teoría de modelos.
- Completitud y categoricidad de teorías.
- Interpolación, preservación y definibilidad.
- Decidibilidad de teorías.
- Estabilidad y teoría de la clasificación.
- Modelos con propiedades especiales (saturados, rígidos, primos,...)
- Teoría de modelos en álgebra.
- Teoría de modelos de estructuras ordenadas: o-minimalidad.
- Modelos de la teoría de conjuntos y de la aritmética.
- Teoría de modelos efectiva y computable.
- Lógica sobre conjuntos admisibles.
- Aplicaciones de la teoría de modelos.

### 4.2 Respaldo y conexión con otros campos

Como puede comprobarse a partir de la enumeración de contenidos previa, la teoría de modelos es un campo extenso con conexiones (y aplicaciones) en otros campos de las matemáticas, desde campos clásicos como álgebra, geometría, análisis o teoría de conjuntos, hasta otros más recientes como el estudio de la complejidad computacional (modelos finitos, aritmética acotada etc.). El departamento de Ciencias de la Computación e Inteligencia Artificial de la Universidad de Sevilla, cuenta con profesorado con una larga trayectoria de docencia e investigación en el campo de la teoría de modelos.