

# Formalización del Razonamiento

Faraón Llorens Largo  
Francisco Escolano Ruiz  
Mar Pujol López  
Otto Colomina Pardo

Grupo de Investigación VGIA (*Visión, Gráficos e Inteligencia Artificial*)  
Departamento de Ciencia de la Computación e Inteligencia Artificial  
Universidad de Alicante  
e-mail: [faraon,sco,mar,otto]@dccia.ua.es

## Resumen

*El presente trabajo analiza las necesidades que tienen los alumnos de la ingeniería en informática en relación con los conceptos relacionados con la lógica y la formalización del razonamiento para posteriormente plantear una propuesta de contenidos distribuidos en las dos asignaturas que el plan de estudios para la Universidad de Alicante ofrece: Lógica de Primer Orden y Ampliación de Lógica.*

## 1. Introducción

El docente que imparte clases en las titulaciones de Ingeniería se enfrenta generalmente a un alumno caracterizado por un disposición eminentemente aplicada, representativo de las carreras tecnológicas. Es por tanto una función primordial de las “asignaturas matemáticas” el pedirles un mínimo de formalización en sus métodos. Cualquier estudiante de Ingeniería debe tener una base matemática sólida que le permita afrontar con éxito otras asignaturas específicas de su titulación así como garantizar un mínimo de rigor y precisión en la aplicación de sus técnicas de trabajo. Estos fundamentos matemáticos deben además estar adaptados a las características propias de la titulación para que de esta forma el alumno reconozca su utilidad, se muestre motivado y no “caigan en saco roto” tanto las nociones transmitidas como las actitudes con las que debe afrontar sus tareas.

El presente trabajo hace un análisis y posterior propuesta de contenidos de *carácter lógico* de la titulación de Ingeniería Informática impartida en la Escuela Politécnica Superior de la Universidad de Alicante. Si queremos ser muy riguroso, podemos decir que nuestros alumnos no tendrán que hacer, en el posterior desempeño de su profesión, demostraciones. Pero, ¿un programa no es un deducción en la cual a partir de unas entradas (*premisas*) debemos obtener unas salidas (*conclusiones*)? ¿la verificación de la corrección de un algoritmo no es una demostración? En apartados posteriores discutiremos más detenidamente sobre la idea de computación como deducción. Además, ideas básicas como recursión, decidibilidad, completitud, coherencia, etc. son fundamentales para entender la computación [1].

## 2. Análisis Previo y Estudio de las Necesidades

Como paso previo a la elaboración de cualquier propuesta se debe analizar el contexto en el que estará inmersa y que, por tanto, ejercerá de marco en el cual debemos encuadrar nuestro trabajo. Dicho marco contextual nos servirá, por una parte, de eje conductor que nos proporcionará las directrices generales pero, por contra, también actuará de corsé al que debemos ceñirnos. En dicha reflexión deberemos contestarnos a una serie de preguntas sobre nuestra tarea de enseñar, ¿qué?, ¿dónde?, ¿cómo? y ¿para qué? Las respuestas a estos interrogantes determinará el diseño final. Una amplia reflexión sobre el tema la podemos encontrar en [2].

Por otro lado, los Planes de Estudios nos proporcionan un marco legal al cual debemos atenemos. Así, para la titulación de Ingeniero en Informática de la Escuela Politécnica Superior de la Universidad de Alicante [3] se establece unas

duraciones y descripciones mínimas, tal como refleja la Tabla 1 : Plan de Estudios de Ingeniero en Informática.

Materia	Curso	Asignatura	Créditos	Teóricos	Prácticos	Contenido
Obligatoria	1	LÓGICA DE PRIMER ORDEN	4'5	3	1'5	Cálculo Proposicional. Lógica de Predicados de Primer Orden
Optativa		AMPLIACIÓN DE LÓGICA	4'5	3	1,5	Lógicas no clásicas. Lógica Difusa. Lógica Temporal. Metalógica

Tabla 1 : Plan de Estudios de Ingeniero en Informática

Este plan de estudios nos limita a 30 las horas teóricas con carácter obligatorio para todos los alumnos que cursen la titulación y a 30 el número de horas optativas. Esto nos obliga a distribuir los fundamentos lógicos que consideramos que deben adquirir los estudiantes de informática en :

- *contenidos comunes*: la Lógica como herramienta básica para el informático, que le servirá para:
  - representar el conocimiento y a formalizar líneas de razonamiento
  - elaborar especificaciones y verificaciones formales de sus programas
  - base teórica de un nuevo estilo de programación: lenguajes de programación lógicos.
- *contenidos específicos*: estudio de otros formalismos lógicos como extensiones de la lógica clásica para aquellos casos en los que esta resulte insuficiente (lógicas “no clásicas”, “desviadas” o “divergentes”).

Para finalizar este primer apartado, estudiaremos las relaciones de dependencia de estas materias con el resto de asignaturas de la carrera. Dicho estudio puede verse resumido de forma gráfica en la Figura 1 : Dependencia entre asignaturas. Aquí únicamente se reflejan las dependencias que aparecen en el Plan de estudios en forma de requisitos. Evidentemente, si analizásemos detenidamente los requerimientos de carácter formal/lógico de las asignaturas encontraríamos un mayor nivel de conexión.

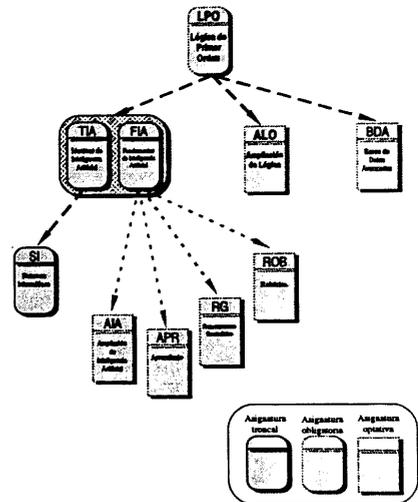


Figura 1 : Dependencia entre asignaturas

Establecido ya el conjunto de supuestos de partida y las metas que deseamos alcanzar, es decir el *currículum*, podemos pasar a plantearnos el diseño de nuestras asignaturas. Tarea que abordamos en el siguiente apartado.

### 3. Nuestra Propuesta

En la Figura 2 podemos ver un mapa de los puntos que consideramos que un ingeniero en informática debe conocer en relación a los formalismos lógicos.

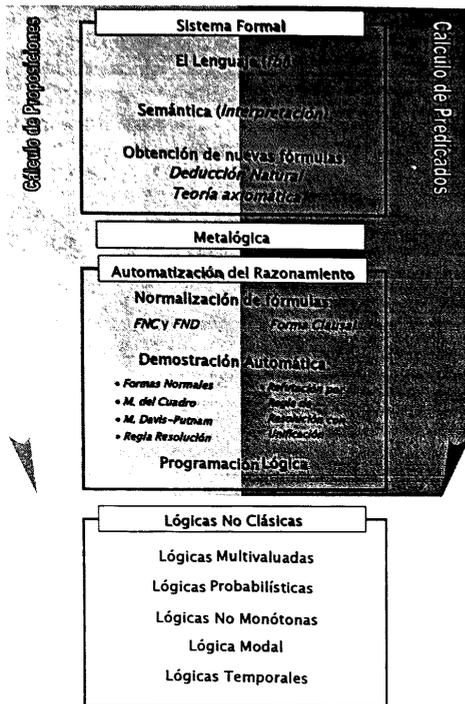


Figura 2 : Mapa de Lógica para Informáticos

Como podemos observar, hemos agrupado los temas en cuatro grandes bloques:

- Sistema Formal
- Metalógica
- Automatización del Razonamiento
- Lógicas No Clásicas

Partiendo de estos puntos clave, pasamos a organizarlos en temas concretos a desarrollar en cada una de las asignaturas, tal y como podemos ver en la Tabla 2 : Contenido de las asignaturas *Lógica de Primer Orden* y *Ampliación de Lógica*. Como podemos observar hay temas que se repiten en ambas asignaturas, ya que en la parte obligatoria se darán a nivel introductorio y en la asignatura optativa se profundizará más en ellos. Mayor detalle del contenido de la asignatura obligatoria *Lógica de Primer Orden* se puede encontrar en [4]. Se ha dejado para la asignatura optativa el tratamiento de los fundamentos formales de los modelos de razonamiento en aquellos dominios en los que la lógica clásica es inoperante o no se ha mostrado lo suficientemente robusta: manejo de la incertidumbre, tratamiento

de la imprecisión, restricciones de tiempo, información incompleta, razonamiento de sentido común, ...

Título	Sesiones
<b>Lógica de Primer Orden</b>	15
1. Historia de la Lógica Simbólica	1
2. El Lenguaje de la Lógica de Primer Orden	2
3. Teoría Semántica	2
4. Deducción Natural	3
5. Teoría Axiomática	1
6. Metalógica	1
7. Normalización de Fórmulas	1
8. Técnicas de Demostración Automática	3
9. Programación Lógica	1
<b>Ampliación de Lógica</b>	15
1. Más allá de la Lógica de Primer Orden:	3
1.1. Cálculo de Predicados con Igualdad	
1.2. Programación Lógica	
1.3. Metalógica	
2. Lógicas Multivaluadas:	3
2.1. Lógicas Trivalentes	
2.2. Lógicas N-valuadas	
2.3. Lógica Difusa	
3. Lógicas Probabilísticas	2
4. Lógicas No Monótonas	2
5. Lógica Modal	2
6. Lógicas Temporales	2
7. Integración de distintas lógicas	1

Tabla 2 : Contenido de las asignaturas *Lógica de Primer Orden* y *Ampliación de Lógica*

Para la parte práctica de las asignaturas, que se desarrolla en los laboratorios informáticos, se ha elegido la línea argumental de la Programación Lógica y en concreto el lenguaje de programación lógica *Prolog*. Aunque este lenguaje se dé en ambas asignaturas, el enfoque será totalmente distinto, ya que se tratará a dos niveles:

- *Prolog como pura lógica* : afrontaremos dos tipos de trabajos prácticos:
  - Base de Conocimientos que contiene el conocimiento relativo a un dominio específico de trabajo junto con las

reglas para obtener nuevas conclusiones

Resolución de Juegos Lógicos, que aportan una doble ventaja, ya que por un lado al tratarse de juegos motivan al alumno, y por otro lado para su resolución necesitan de un proceso deductivo.

- *Prolog como lenguaje de programación* : el acercamiento al concepto declarativo de la programación nos permite aproximar las nociones de programa y especificación, pero buscando eficiencia en su ejecución, para lo que disponemos de una serie de predicados metalógicos. Por otro lado, también se trabajará con variaciones del lenguaje Prolog: Prolog++, dProlog, fProlog, ...

Tal como comentan sus autores en la introducción del libro [5] “este lenguaje es, a la vez, fácil y difícil”, ya que en poco tiempo podemos escribir programas con resultados espectaculares, pero elaborar programas complejos requiere un dominio conceptual del lenguaje laborioso de adquirir. Nuestra propuesta es aprovechar esta dualidad : para en la asignatura obligatoria de primer curso explotar la cara fácil de Prolog, de forma que rápidamente puedan escribir bases de conocimiento que razonen correctamente así como resolver problemas lógicos con pequeños programas (explotando la potencia declarativa del lenguaje); para la asignatura optativa el alumno deberá profundizar en la potencialidad de Prolog como lenguaje de programación y conocer las técnicas de dicho estilo de programar.

#### 4. Características Esenciales

La justificación de nuestra propuesta vendrá dada por las siguientes característica que queremos destacar de ella y que consideramos importantes:

- *Lógica para Informáticos*: La Lógica se ha convertido en uno de los fundamentos matemáticos y en una base formal indispensable en todo informático. La formalización del conocimiento y la automatización de las formas de razonamiento son primordiales en muchas áreas de la

Informática. La importancia de la Lógica en los curricula de Informática va tomando cuerpo propio debido a sus aplicaciones en contextos específicos tales como la Programación, la Ingeniería del Software, el Diseño de Sistemas de Bases de Datos y la Inteligencia Artificial, entre otros.

- *La Computación como una forma de Deducción*: un algoritmo (y por extensión un programa de ordenador) lo podemos ver como una deducción en la cual nuestras premisas son los parámetros de entrada y la conclusión que queremos deducir son los datos de salida; las reglas que podemos utilizar serán el conjunto de instrucciones que nos proporciona el lenguaje.
- *Programación Lógica*: un objetivo básico debe ser el estudiar la aplicación de la lógica a la resolución de problemas y a la programación de ordenadores. Utilizaremos como base la Forma Clausal, que siendo más sencilla es igual de poderosa, y la interpretación de inferencia lógica como resolución de problemas. Empleada como un lenguaje para comunicarse con los ordenadores, la lógica representa un formalismo de nivel superior y más orientado a la persona que otros lenguajes de programación. Toda esta base teórica dio como resultado la aparición de una clase de lenguajes de programación: la Programación Lógica.
- *Metalógica*: utilizando las palabras de Hofstadter, “cuando se estudian sistemas formales, es importante distinguir entre lo que se hace *dentro* del sistema, por un lado, y por otro las enunciaciones u observaciones que se formulan *acerca* del sistema”<sup>1</sup>. Debemos ascender un nivel (de ahí el prefijo *meta*) para contemplar la lógica en su conjunto y así estudiar las propiedades de los cálculos lógicos (consistencia, completitud, y decibilidad, entre otras).
- *Lógicas No Clásicas*: posiblemente el rasgo más destacable del estado actual de la Lógica sea la existencia y proliferación de las llamadas lógicas no clásicas, surgidas como respuesta a la aplicación de la lógica a la Inteligencia Artificial. Se trata de sistemas lógicos que difieren en una o más cualidades de la lógica clásica, por lo que son vistas como extensiones a la lógica “normal”, calificándolas así como lógicas “desviadas” o “divergentes”.

<sup>1</sup> Gödel, Escher, Bach, Douglas Hofstadter, página 43.

Estas líneas generales pueden ser complementadas, tanto en metodología docente como en principios básicos, con las características propuestas en [6] para la asignatura *Lógica de Primer Orden*:

- La lógica es divertida
- Nuestros alumnos son informáticos
- La mejor manera de aprender un lenguaje es usándolo
- La práctica como apoyo a la teoría
- Introducción a la Programación Lógica
- Personalización del contenido
- Diversidad en la evaluación

## 5. Conclusiones

Cuando un docente se plantea la programación de una propuesta para una determinada asignatura, inmediatamente se verá desbordado por la cantidad de saber que encierra e irremediamente deberá utilizar la "tijera" y recortar sus pretensiones, ya que su tarea también está limitada en el tiempo (en nuestro caso un cuatrimestre). El problema estriba en decidir por qué sí y por qué no, en determinar qué incluyo y qué me dejo fuera, en establecer a qué nivel profundizo o si me quedo en la superficie, ...

Este trabajo pretende contestar a los interrogantes antes planteados y justificar las decisiones tomadas. Como siempre, esto no es la última palabra y a la finalización de cada curso escolar deberemos reflexionar sobre nuestro trabajo y modificar los puntos flojos, actualizar

con temas novedosos y potenciar los aspectos favorable.

## Referencias

- [1] D. R. Hofstadter. *Gödel, Escher, Bach. Un Eterno y Grácil Bucle*. Tusquets Editores, 6ª edición, 1998
- [2] F. Llorens. *Proyecto Docente (Lógica de Primer Orden, Ampliación de Lógica y Fundamentos de la Programación)*. Profesor Titular de Escuela Universitaria en el área de Ciencia de Computación e Inteligencia Artificial, Universidad de Alicante, 1996
- [3] *Plan de Estudios conducente al título de Ingeniero en Informática de la Escuela Politécnica Superior de la Universidad de Alicante*. Resolución de 18 de septiembre de 1992 de la Universidad de Alicante. B.O.E. nº 37, de 12 de febrero de 1993, pag. 4467-4478.
- [4] F. Llorens y Mª J. Castel. *Lógica de Primer Orden*. Servicio de Publicaciones, Universidad de Alicante, 1998 (en prensa).
- [5] F. Giannesini y otros. *Prolog*. Addison-Wesley Iberoamericana, 1989.
- [6] F. Llorens y Mª J. Castel. *Lógica de Primer Orden en las Ingenierías Informáticas*. II Jornadas Nacionales de Innovación en las Enseñanzas de las Ingenierías, I.C.E. Universidad Politécnica de Madrid, 1996.