CSI: Presentación

Javier Larrosa

UPC Barcelona Tech

.

Motivación

- En muchos contextos se necesita un sistema automático que **aprenda** y/o **razone**
  - **Aprender** = Identificar e incorporar información relevante en el **contexto** de nuestro sistema
  - **Razonar** = Usar la información disponible para establecer ...
  - ...qué puede ser cierto o qué es necesariamente cierto con la información que tenemos
  - ...cómo actuar de manera óptima con la información que tenemos
- Esto puede ser tanto en,
  - Sistema Autónomo (robots, trading software, coches autónomos,...)
  - Sistema de ayuda (Diagnosis, Recomendadores, ...)

Descripción del curso

- **Contentido (3 partes x 4 semanas):**
  - Constrait Programming
  - Bayesian Networks
  - Machine Learning
- **Orientación:** Fuerte componente práctica pero de propósito general
- Las técnicas que veremos, y otras relacionadas, tienen muchos campos de aplicación. Algunas de gran actualidad.
- **Evaluación:** En cada una de las tres partes haremos: Examen (E) y una o varias prácticas (P). Sobre una nota final de 100, los exámenes valen 20 puntos y las prácticas valen 13 puntos.

Este último punto que falta nos lo dan gratis.

Aproximación Declarativa

- Tenemos un **Lenguaje de Modelado** (representacion del conocimiento)
- Tenemos una librería de **Algoritmos** (solvers) que *entienden* el lenguaje
  Caja negra que entiende el lenguaje de modelado, es muy compleja.
- Con el lenguaje, modelamos (manual o automáticamente) el sistema sobre el que queremos razonar
- Con los algoritmos obtenemos respuesta a preguntas de interés sobre el sistema Son algoritmos que resuelven problemas
- **Separación entre modelo y razonamiento**
- Desde un punto de vista matemático, casi siempre se reduce a optimizar una funcion objetivo:

$$x^* = \max_{x \in X} F(x)$$

Eugene Freuder 87

Constraint Programming represents one of the closest approaches computer science has yet made to the **Holy Grail of programming**: **the user states the problem, the computer solves it**.

Básicamente tenemos una función objetivo y unas restricciones.

Objectivo

Objetivo

Aprender a identificar, modelar y resolver **problemas de optimización discreta**

Usaremos un lenguaje llamado **MiniZinc** (www.minizinc.org)

- Es un lenguaje de modelado
- Independiente de los algoritmos que haya por debajo
- Realmente es un lenguaje de modelado + compilador a lenguajes más básicos.
- Solvers posibles:
  - CPLEX, Gurobi, Gecode, OR-tools, Oscar,...

Por qué MiniZinc

MiniZinc

MiniZinc es muy pedagógico (intuitivo, bien documentado,...), pero hay otras alternativas

Alternativas Actuales:

- CPMpy, PyCSP,...
- GAMS, AMPL, AIMMS, XPRESS-MP ...