Tema 2

Técnicas informáticas de aplicación en logística

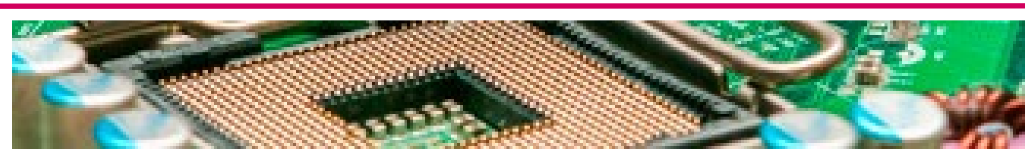
Logística y Servicios

Un **problema** de gestion y, **sobre todo**, de **optimización**

Retos a resolver

Desde el punto de vista del **inventario de una empresa**

- ¿Cómo se gestiona el inventario según el tipo de demanda de la empresa? ¿Por qué resulta fundamental gestionar bien el inventario para tener un buen nivel de servicio al cliente?
- ¿Cuáles son los principales costes asociados?
- ¿Qué cantidad hace falta pedir y cuándo se pide?
- ¿Qué es el lote económico y para qué sirve? ¿Qué método de lotificación podemos aplicar ante una demanda variable?
- ¿Qué es el periodo económico y para qué sirve?
- ¿Cuáles son los principales enfoques de gestión de inventario actuales?



Desde el punto de vista de la **localización** en la empresa

- ¿Dónde se deben posicionar los nodos (almacenes, centros de distribución, centros de venta, etc.)?
- ¿Cuáles son sus costes y capacidades máximas?
- ¿Cuánta demanda pueden cubrir (capacidad de cubrimiento)?
- ¿Cómo se distribuyen los recursos para atender a la producción de la empresa y cuál es su coste?
- ¿Cuál es la planificación del aprovisionamiento?
- ¿Cómo son las redes de interconexión?

Desde el punto de vista del **transporte** y **distribución**

- ¿Cómo se debe diseñar la red de distribución?
- ¿Cuál es la planificación de la distribución?
- ¿Cómo se calculan/planifican/optimizan las rutas de transporte? ¿Cuáles son los costes?
- ¿Se utilizarán medios de transporte propios o se pueden subcontratar? Reubicación de vehículos
- ¿Cómo se distribuyen las mercancías? ¿Existen distintos tipos con distintas características (peso, volumen, prioridad, perecederos, frágiles, restricciones de frío, etc.)?

Hay múltiples alternativas de **decisión** (posiblemente a optimizar)

Problema real → Modelado → Modelo → Técnicas y herramientas → Solución

Planificar, asignar, optimizar, etc. En general, **decidir**

Formulación matemática o aproximada, descripción en un lenguaje específico, etc.

Resultado: asignación de valores (personal, recursos, horarios, rutas, etc.)

Problema real $\xrightarrow{\text{Modelado}}$ Modelo $\xrightarrow{\text{Técnicas y herramientas}}$ Solución

1. Decisión **determinista**: con resultados y probabilidades conocidas, sin riesgo, ni incertidumbre, ni contrarios

- Problema "simple" de toma de decisiones. Herramientas específicas
- Investigación operativa
- Satisfacción de restricciones. Planificación/Scheduling
- Búsqueda en general

Problema real → (Modelado) → Modelo → (Técnicas y herramientas) → Solución

1. Decisión **determinista**: con resultados y probabilidades conocidas, sin riesgo, ni incertidumbre, ni contrarios

Se abordan mediante:

- Métodos exactos (matemáticos y admisibles)
- Métodos aproximados (heurísticas y meta-heurísticas)
- Optimización mono- o multi-objetivo

$$\text{Problema real} \xrightarrow{\text{Modelado}} \text{Modelo} \xrightarrow{\text{Técnicas y herramientas}} \text{Solución}$$

2. Decisión **no determinista**: los resultados no están completamente determinados

- Decisión con riesgo: se conocen las utilidades y probabilidades de los resultados de cada decisión → lotería
- Decisión con incertidumbre: se desconoce la probabilidad del resultado de cada decisión →múltiples criterios (arriesgados, conservadores, optimistas, pesimistas, etc.) Árboles y redes de decisión

Problema real $\xrightarrow{\text{Modelado}}$ Modelo $\xrightarrow{\text{Técnicas y herramientas}}$ Solución

3. Decisión **ante oponentes/contrarios** (se desconoce lo que puede hacer el oponente ante nuestra decisión)

- Teoría de juegos
    - Estrategias dominantes
    - Estrategias de equilibrio (Nash)
    - Estrategias ganadoras
    - Estrategias reactivas

Sistemas **ERP** (Enterprise Resource Planning) de planificación integrada de recursos y gestión empresarial

Incluyen módulos de inventario, producción, recursos humanos, distribución y facturación

- SAP. Gestor de recursos empresariales con distintos módulos. Ej. Warehouse Management
- Sage 200, Odoo, Syspro, etc.

Módulos específicos en herramientas. Ej. Complemento Solver de Excel

Tema 3

Optimización matemática

Obtener el valor de las variables de decisión, tal que se cumplan las restricciones y se maximice (o minimice) una determinada función objetivo

Dado un conjunto de variables V, un dominio combinado X $(|v1| \otimes |v2| \otimes \ldots\ldots \otimes |v1|)$, un conjunto de restricciones C entre las variables, y una función objetivo:

$$f(x): \; x \in X \;\rightarrow\; \Re \qquad \textit{;x: asignación valor a variables}$$

El objetivo es obtener x*, que cumpla las restricciones C, y:

$$x \in X: \quad f(x^*) \geq f(x), \;\; \forall x \in X$$

Optimización combinatoria: Variables discretas

Optimización matemática (complejidad NP en peor caso)

- Programación Lineal (LP). Método Simplex de Investigación Operativa

Modelización del problema mediante un conjunto de **inecuaciones lineales**, con un objetivo también **lineal**. Las variables son **números reales** mayores o iguales a cero.

Variables: $x_i \geq 0$, $x_i \in \Re$

Conjunto de restricciones , expresión lineal de la forma: $\sum_{i=1,n} b_{ij} x_i \quad \leq | = | \geq \quad C_j$,

Función objetivo: $f = Max\ (\sum_{i=1,n} a_i x_i\ )$, $x_{1..n} \in$ Conjunto-Factible

Tema 4 y 5

Optimización matemática (complejidad NP en peor caso)

- Programación Lineal entera-mixta (MIP)

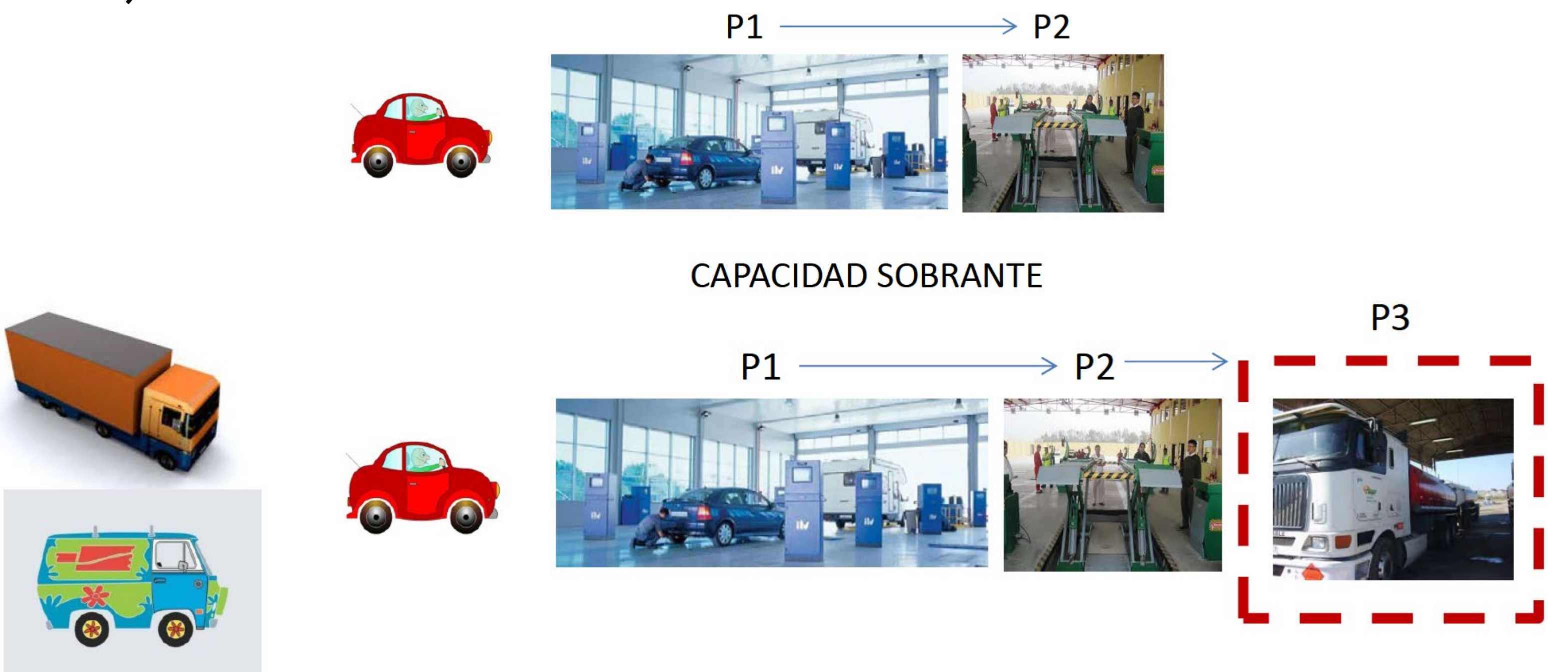
Algunas variables solo pueden tomar valores enteros.
Problema de Optimización Combinatoria. Complejidad NP.
Método típico: Branch and Bound (incluyendo relajación, ajuste de cotas, ramificación y poda, etc).
Herramientas: GAMS , CPLEX, Lingo, etc.
Caso particular: **Programación binaria** $x_i \in \{0, 1\}$. Típico en problemas de asignación.

- Programación no-lineal (NLP), donde alguna de las restricciones o la función objetivo no son lineales (ej. ecuaciones cuadráticas o cúbicas, definición de curvas, etc.)

Tema 4 y 5

Ejemplo. Decisión en logística de servicios

Se quiere ampliar una línea de inspección de vehículos de dos etapas {P1, P2} infrautilizadas, con una tercera etapa P3, a fin de poder inspeccionar nuevos tipos de vehículos (furgonetas y camiones)



Ejemplo. Decisión en logística de servicios

La capacidad sobrante al día (en minutos) de P1 y P2 se indican en el cuadro, así como el tiempo requerido (en minutos) por cada nuevo tipo de vehículo en dichas etapas.

La nueva etapa será rentable siempre que se utilice un mínimo de 500 minutos al día.

Por otra parte, la inspección de cada tipo de vehículo supone un determinado coste en material fungible indicado en la tabla, no queriendo sobrepasar un coste diario total de 300 euros.

Suponiendo que se inspeccionarán 3 furgonetas por cada camión, ***se desea estimar el máximo número de cada tipo de vehículo que podría ser inspeccionado en el nuevo servicio.***

	Ocupación P1	Ocupación P2	Ocupación P3	Coste Fungible
Capac. Sobrante	150’	180’		
Furgoneta	2’	4’	8’	2
Camión	3’	3’	12’	5

Ejemplo. Decisión en logística de servicios

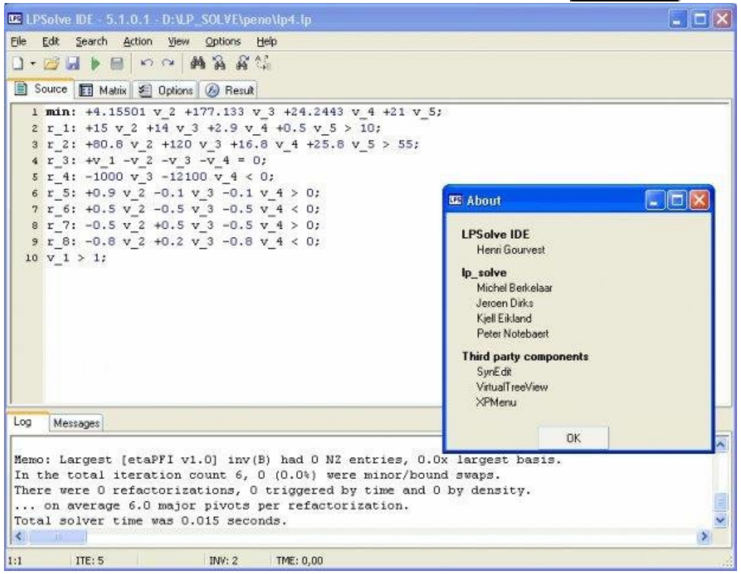
	Ocupación P1	Ocupación P2	Ocupación P3	Coste Fungible
Capac. Sobrante	150’	180’		
Furgoneta (3x)	2’	4’	8’	2
Camión (1x)	3’	3’	12’	5

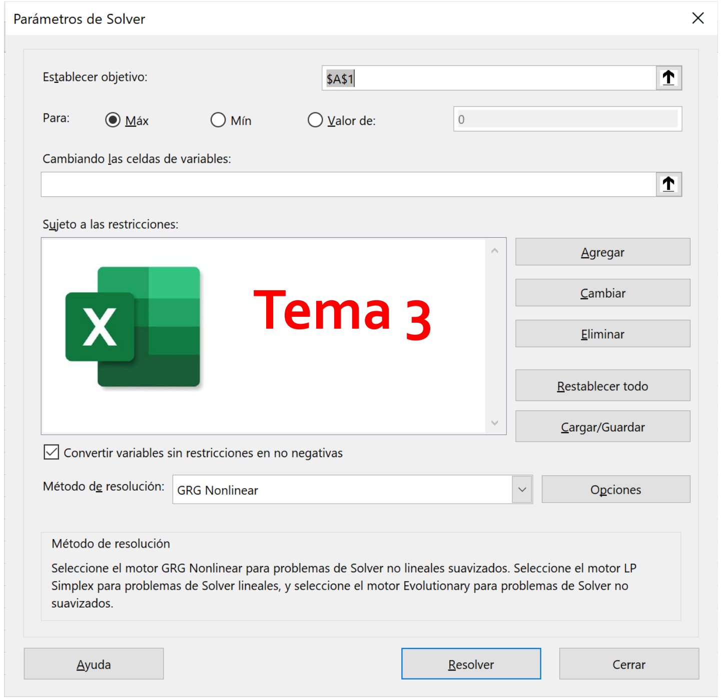
Ejercicio

Ejemplo. Decisión en logística de servicios

Se puede resolver de múltiples formas: Simplex, Lingo, Lindo, lpsolve, GAMS, ILOG CPLEX, Excel, etc.

LINDO

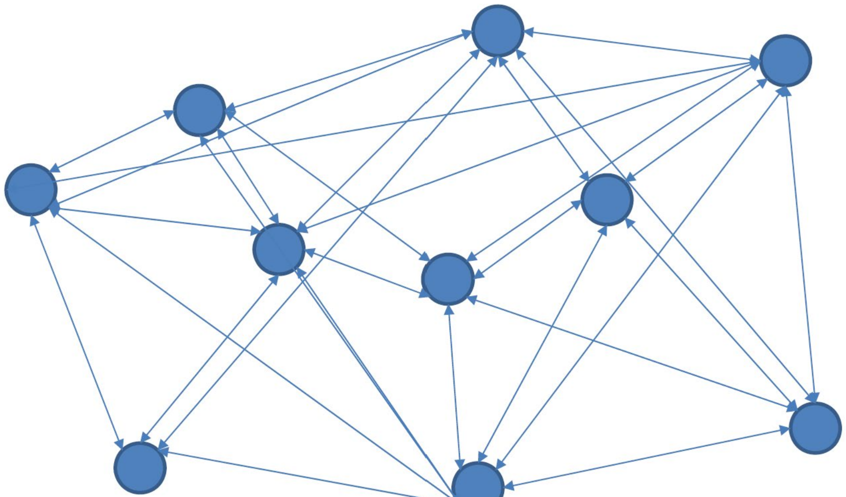




Ejemplo. Localización de almacenes de distribución a nodos consumidores

Dados distintos Nodos_i consumidores *(i=1..n)*, se quiere obtener el número, capacidad y situación de posibles Centros_j de distribución *(j=1..m)*, adjuntos o no a los nodos, minimizando su construcción y el coste de transporte desde cada centro de distribución a cada nodo consumidor.
El coste del transporte depende de la distancia y producto transportado. El producto a distribuir es único.

Red de comunicación (conexión entre nodos)



Ejemplo. Localización de almacenes de distribución a nodos consumidores

DATOS:

a_i = Vector de demandas del Nodo consumidor *i*
d_{ij} = Matriz de distancias entre el Nodo de demanda *i* y el Almacén de distribución *j*.
f_j = Vector de coste de construcción de un Almacén de distribución en *j*
c_{ij} = Matriz de coste de transporte por unidad de producto y unidad de distancia
Cj = Vector de capacidad del Almacén de distribución si se ubica en *j*

VARIABLES

X_{ij} = Matriz de demanda del Nodo *i* atendida por los productos previstos en el Almacén *j*
W_j = 1, si se ubica un almacén de distribución en *j*; 0, en caso contrario.

Las variables W_j son enteras (típico problema de asignación {0, 1}): La decisión es {T, F}.

Satisfactibilidad y Optimización en IA (complejidad NP)

- Satisfacción de restricciones (Constraint Satisfaction Problem, CSP) y Optimización de Restricciones (COP)

Un **Problema de Satisfacción de Restricciones (CSP)** se representa como:

1. Conjunto de **Variables**: $\mathbf{X} = \{x_1, x_2, ..., x_n\}$
2. **Dominios** de Interpretación para las variables: $\mathbf{D} = \{D_1,...,D_n\} / x_i \in D_i$
3. Conjunto de **Restricciones** entre las variables: $\mathbf{C} = \{c_1, c_2, ..., c_m\}$

Las restricciones expresan las combinaciones válidas de valores entre las variables. Puede existir una métrica a optimizar

Satisfacción/optimización de restricciones (CSP y COP)

Tipos de restricciones:

- Restricción Unaria (para una variable). Ej: $a \leq x_i \leq b$, $a,b \in Z$
- Restricción Binaria (entre dos variables x_i, x_j). Ej: $a \leq p_i x_i + p_j x_j \leq b$, $a,b,p_i,p_j \in Z$
- Restricción no-binaria (n-aria, entre n variables). Ej: $a \leq p_i x_i + p_j x_j + \ldots + p_n x_n \leq b$, $a,b,p_i,p_j \ldots p_n \in Z$
- En general: $\sum_{i=1}^{n} p_i x_i \ \{<, \leq, =, \neq, >, \geq\}\ b$
- Una **solución** es una instanciación consistente de todas las variables, tal que todas las restricciones del CSP se cumplan (y **optimicen** la métrica si existe en un COP)

Ejemplo. Dos opciones de transporte

El transporte de un **paquete1** se puede realizar en furgoneta (30-40 minutos) o en camión (al menos una hora). El transporte de un **paquete2** puede ser también en furgoneta (20-30 minutos) o en camión(40-50 minutos)

El **paquete1** ha salido entre las 8:10 y las 8:20 y el **paquete2** ha llegado entre las 9:00 y las 9:10.

Además, sabemos que el **paquete1** ha llegado entre 10 y 20 minutos después de que el **paquete2** saliera

¿Es esto posible?

Ejemplo. Dos opciones de transporte

Ejercicio

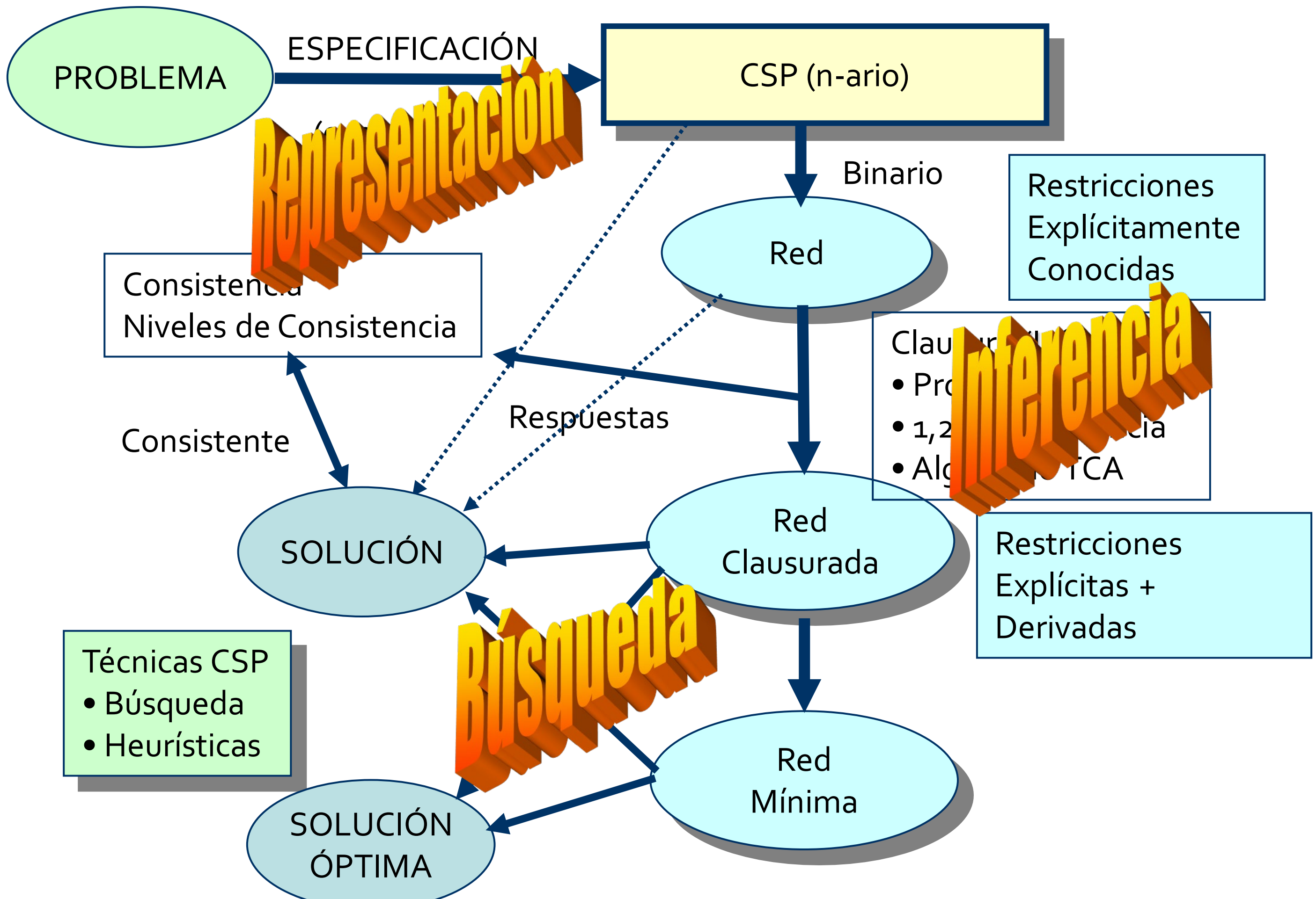
Satisfacción/optimización de restricciones (CSP y COP)

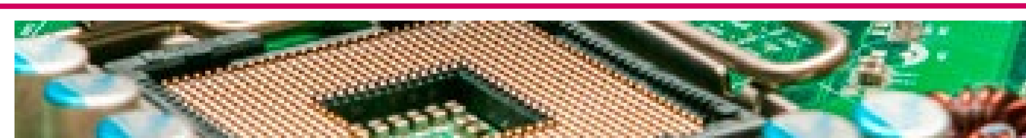
Se puede resolver de múltiples formas: MiniZinc, Choco, Google OR-Tools, ILOG CPLEX, ECLiPSe Constraint Programming System, CP-SAT Solver, etc.

CHOCO **Tema 7**

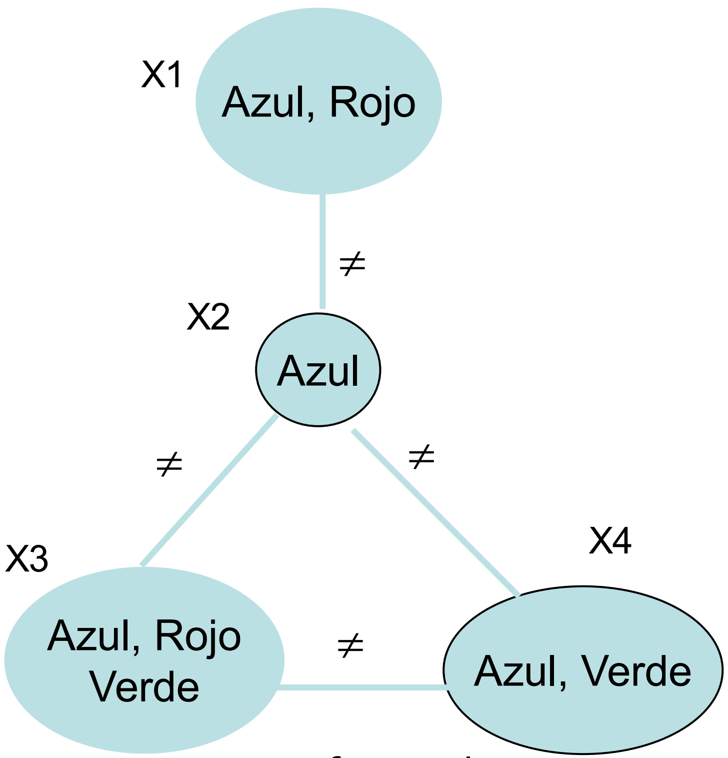
Repositorio: http://www.constraintsolving.com

Satisfacción de restricciones (CSP) y **técnicas de clausura**





Satisfacción de restricciones (CSP). **Ejemplo** de clausura. ¿Cómo eliminar valores de dominios con 2-consistencia?



Grafo Inicial

Ejercicio

Satisfacción de restricciones (CSP). **CSPs flexibles. Ejemplos**

CSP Posibilista. Permite que **algunas restricciones no se cumplan**, lo que conlleva un coste de insatisfabilidad.

Restricción $\text{Soft}_k \Rightarrow \text{Coste}_k$ asociado a su insatisfabilidad

El objetivo es obtener una solución que **minimice el coste de las restricciones no satisfechas (COP)**

Satisfacción de restricciones (CSP). **CSPs flexibles. Ejemplos**

CSP Ponderado (Weighted CSP). Existen restricciones disyuntivas ponderadas (o valores posibles de las variables). Cada disyunción tiene un coste asociado a su satisfactibilidad.

$$\text{Restricción Soft} \Rightarrow C_1\,(\text{Coste}_1) \vee C_2\,(\text{Coste}_2) \vee \ldots \vee C_k(\text{Coste}_k)$$

El objetivo es obtener una solución que **minimice el coste de las restricciones satisfechas (COP)**

Planificación inteligente

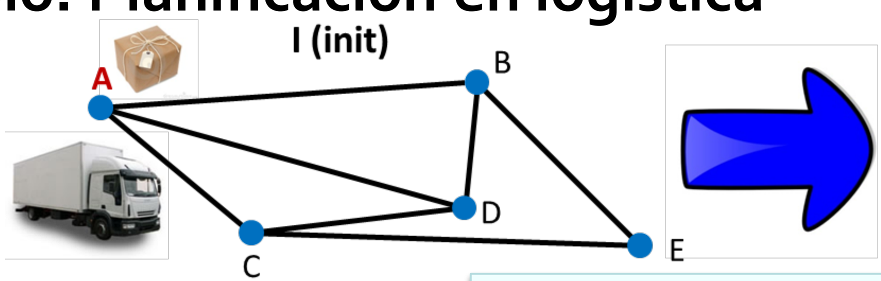
- Proceso de **razonamiento** inteligente para seleccionar y organizar un conjunto de acciones que permita la consecución de unos objetivos en base a su **causalidad**

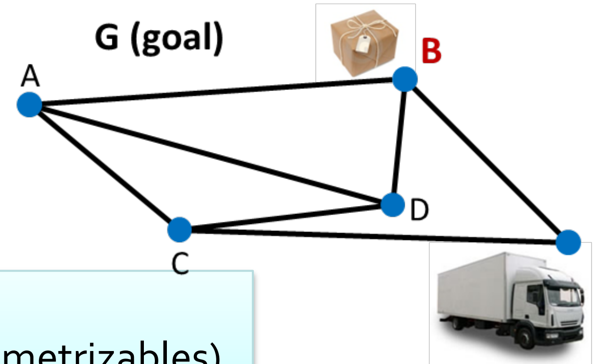
- Proceso **deliberativo** en el que tratamos de anticiparnos a los efectos de las acciones, por lo que es **previo** a la ejecución (actuación) de las acciones

Problema de planificación

- **Modelado** describe
  - un conjunto de acciones (¿**qué** se puede hacer?),
  - un estado inicial del entorno (¿**desde dónde**?),
  - una descripción del objetivo a conseguir (¿**hasta dónde**?)
- **Resolución** obtiene
  - acciones + órdenes de ejecución para pasar del estado inicial en un estado que satisfaga el objetivo. Decide qué acciones planificar

Ejemplo. Planificación en logística





Objetos (Variables de Estado)
Camión, Paquete

Predicados (estados de los objetos):

{camión, paquete} **en** {A,B,C,D,E}
{paquete} **dentro** {camión}

Problema concreto que se define sobre el dominio

Predicados Proposicionales:

I (en camión A)
(en paquete A)
G (en camión E)
(en paquete B)

Acciones del Dominio
(utilizan plantillas parametrizables)

A **move** (?cam - camion, ?origen - city, ?destino- city)
pre: (en ?cam ?origen)
add: (en ?cam ?destino)
del: (en ?cam ?origen)

load (?paq - object, ?cam - camion, ?lugar - city)
pre: (en ?paq ?lugar)
(en ?cam ?lugar)
add: (dentro ?paq ?cam)
del: (en ?paq ?lugar)

unload (?obj - object, ?cam - camion, ?lugar - city)

Modelado en PDDL (Planning Domain Definition Language)

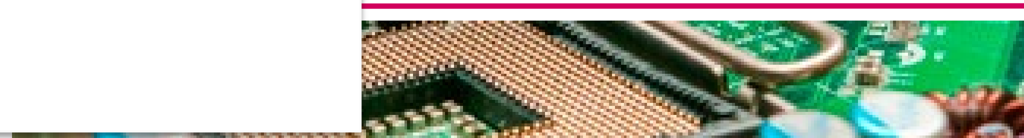
```
(define (domain logistics)     ; Define los tipos, predicados y operadores
(:requirements :strips :typing)
(:types     ; tipos utilizados
      package  location  vehicle  - object
      truck  - vehicle
      city  - location)
(:predicates
    (at  ?vehicle-or-package  - (either vehicle package)   ?location - location)
    (in  ?package - package    ?vehicle - vehicle))

(:action load       ;acción para cargar
 :parameters (?obj - package    ?truck - truck    ?loc - location)
 :precondition  (and  (at ?truck  ?loc)
                      (at  ?obj  ?loc))
 :effect  (and (not  (at  ?obj  ?loc))        ;efecto delete
                (in  ?obj  ?truck)))          ;efecto add

(:action unload      ;acción para descargar
 :parameters   (?obj - package   ?truck - truck    ?loc - location)
 :precondition  (and  (at  ?truck  ?loc)
                      (in  ?ob j ?truck))
 :effect  (and  (not  (in  ?obj  ?truck))
                (at  ?obj  ?loc)))

(:action move ...     ;acción para mover
```

```
(define (problem pb1)    ; Objetos, Inicial  y Objetivo
(:domain logistics)
(:objects  pck1- package
           pck2 - package
           pck3 - package
           truck1 - vehicle
           truck2 - vehicle
           madrid - city
           valencia - city
           barcelona - city ...)
(:init     (at truck1 valencia) (at truck2 madrid)
           (at pck1 barcelona) (at pck2 madrid)...)
(:goal     (and  (at pck1 madrid)
                 (at pck2 valencia)
                 (at truck1 madrid)
                 (at truck2 barcelona)))
```



Proceso de planificación

Especificación PDDL
<Dominio> + <Problema>

Modelado (PDDL)

PDDL + LPG
Tema 6

RESOLUTOR (planificador)

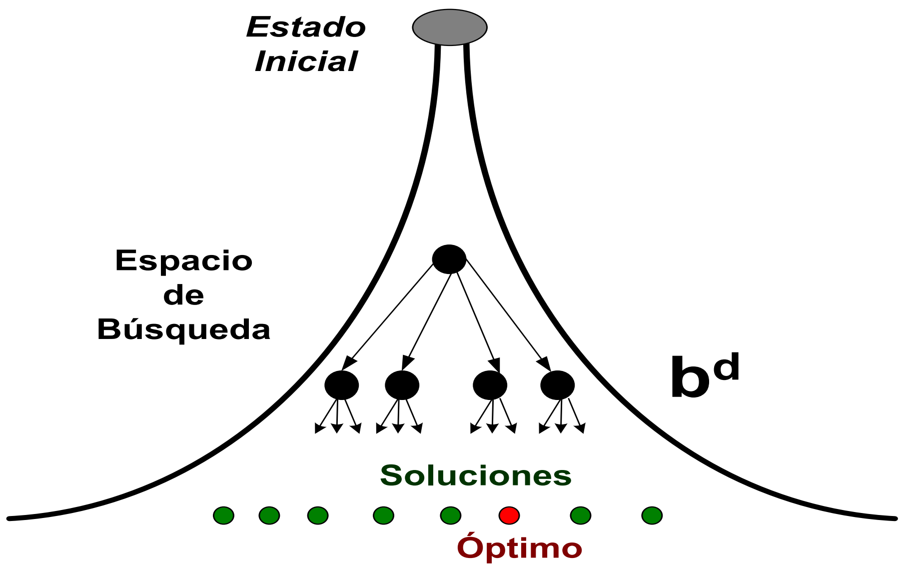
Plan

Resolución mediante **búsqueda** (puede incluir optimizar una métrica)

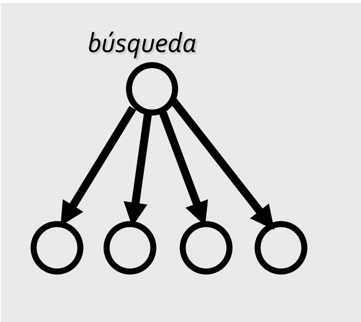
- En un espacio de estados o de planes
- Métodos heurísticos (relajación, abstracción, etc.)

En general, la **búsqueda de soluciones** factibles/óptimas es un problema difícil de resolver (NP-completo en satisfactibilidad y NP-duro en optimalidad)

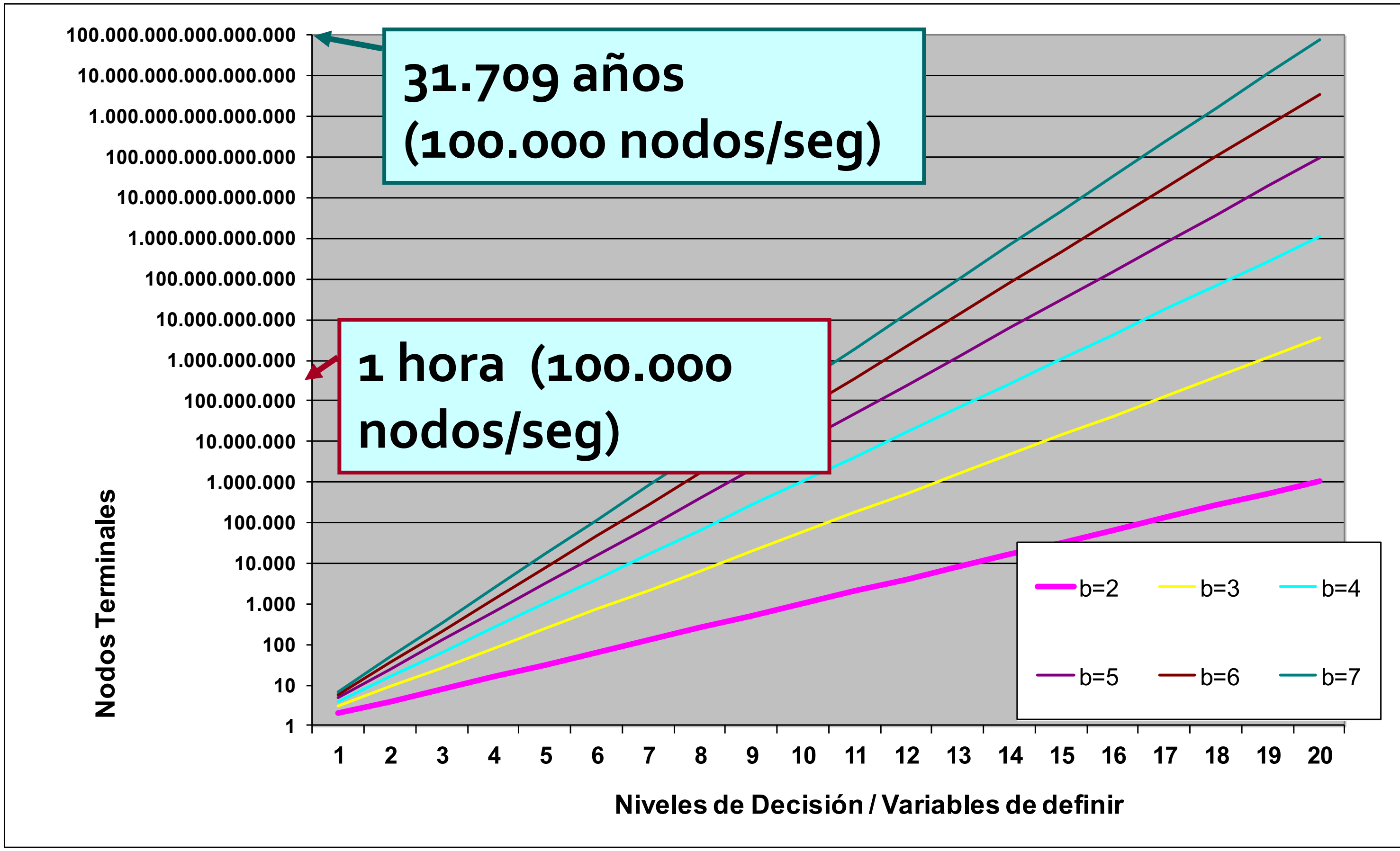
En problemas **combinatorios** no se puede garantizar encontrar la solución óptima en un tiempo computacional razonable



Existe una **explosión combinatoria**



Factor de ramificación (b) y nivel de profundidad o decisión (d)

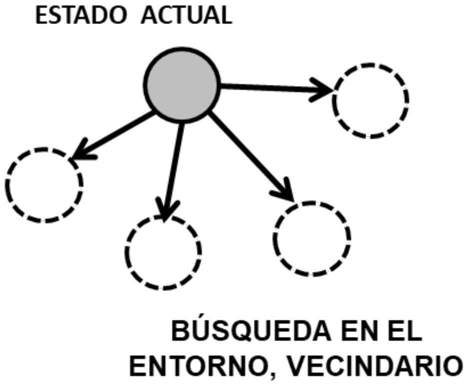


Búsqueda Local (local search):

En cada iteración, se busca solo en el ***entorno del estado actual***.

- ***Eficientes en memoria (solo estado actual y sucesores)***
- ***Coste temporal sigue siendo exponencial.***
- ***Funciones de evaluación*** para elegir el mejor estado sucesor.
- ***Inconvenientes:***

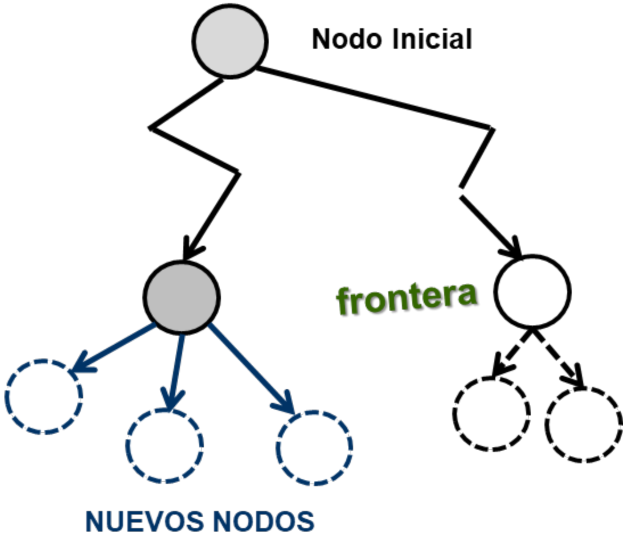
  Pueden quedar atrapadas en soluciones que no admiten mejoras en su entorno (óptimos locales, valles, mesetas), no garantizan óptima, problemas de incompletitud, ciclos, etc.



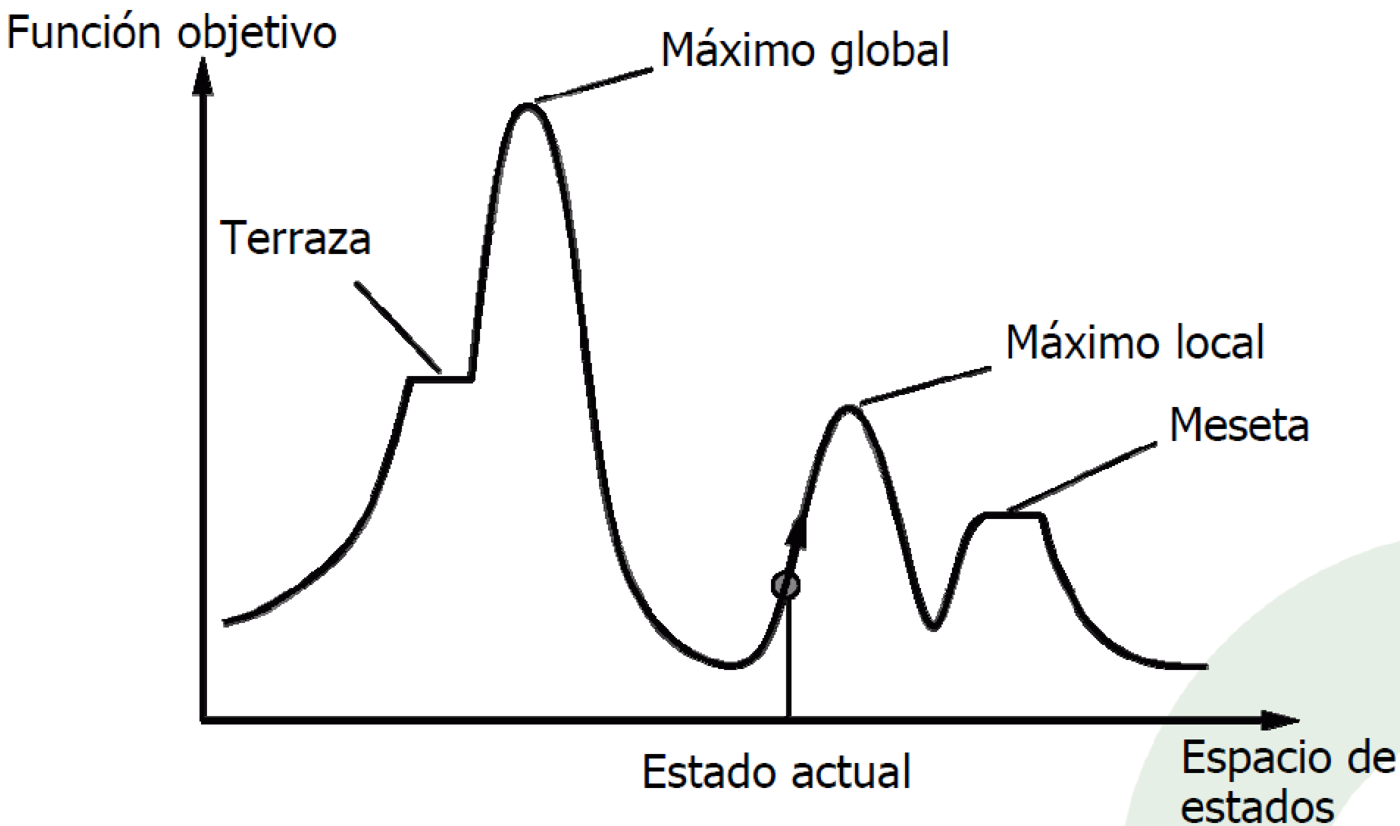
Búsqueda Global (global search):

Búsqueda sistemática en todo el espacio de búsqueda.

- Alto coste en memoria (*explosión combinatoria*).
- Puede ser completa y admisible (garantizar solución óptima)
- Suelen hacer uso de ***funciones heurísticas***.



A la hora de optimizar, la búsqueda **local** puede quedarse atrapada en un **óptimo local** – búsqueda incompleta



A la hora de optimizar, la búsqueda **global** es **sistemática** en el espacio de búsqueda (pero cara) – mantiene abierta toda la frontera

Se define una **función de evaluación f(n)** para evaluar cada nodo del árbol de búsqueda. Típicamente, $f(n)=g(n)+h(n)$

- **Coste uniforme**: $f(n)=g(n)$. La primera solución es la óptima, pero complejidad muy alta
- **Búsqueda voraz**: $f(n)=h(n)$. Búsqueda típica de primero el mejor. Complejidad menor, pero no garantiza que la primera solución sea la óptima

Algoritmo A, que mantiene la doble evaluación: $f(n)=g(n)+h(n)$

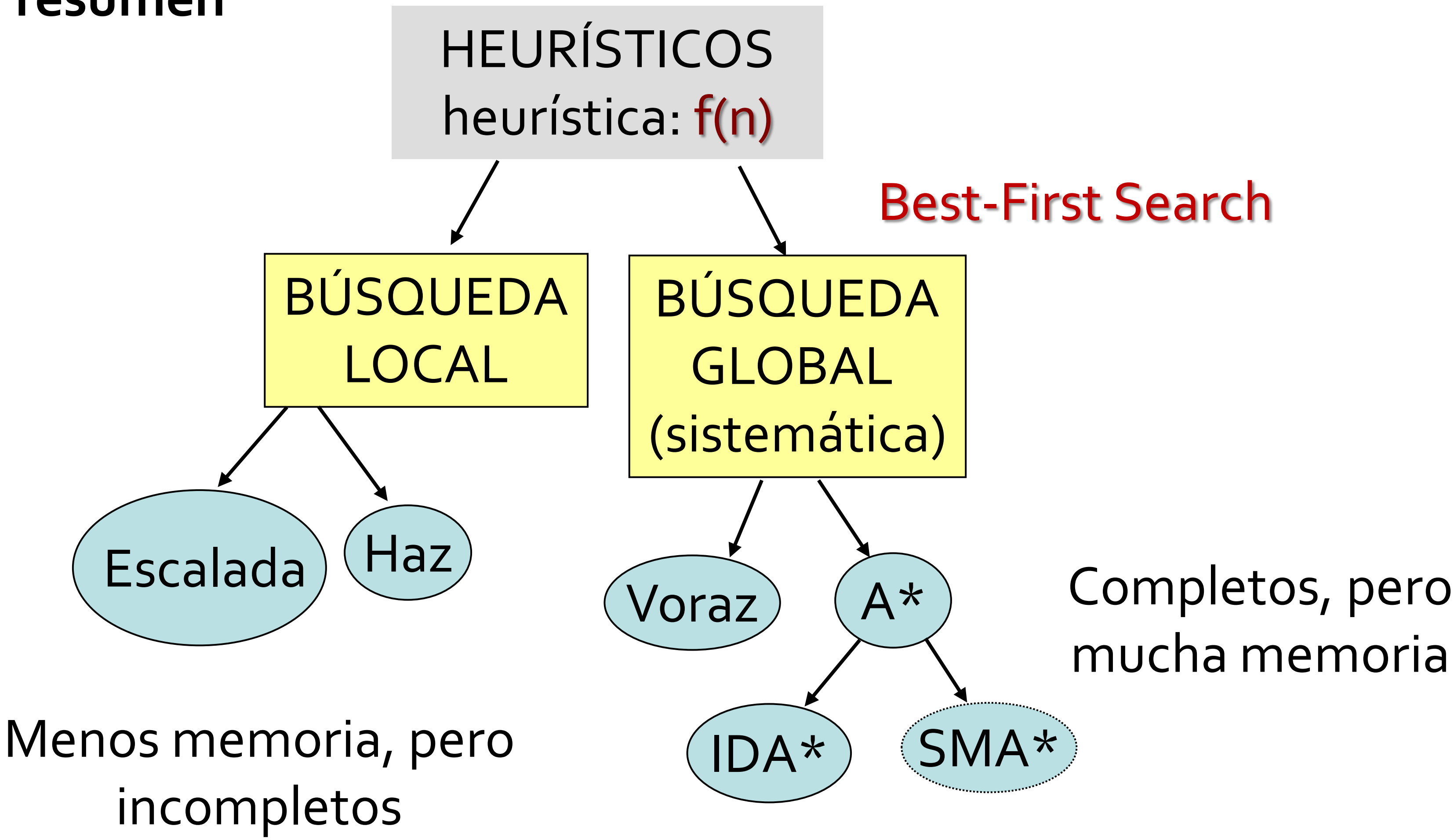
Combinación de:

- Coste uniforme. Completa y óptima, $g(n)$, pero ineficiente.
- Búsqueda voraz. Reduce coste de búsqueda por $h(n)$, pero no garantiza optimalidad

Algoritmo A*: $\forall n,\ h(n) \leq h^*(n)$

- Completa y admisible. La primera solución es óptima
- Requiere mucha memoria (un algoritmo que expanda menos nodos que A* no garantizará admisibilidad)
- Con variantes: A* ponderado, IDA*, SMA*

En **resumen**


HEURÍSTICOS
heurística: f(n)
Best-First Search
BÚSQUEDA LOCAL
BÚSQUEDA GLOBAL (sistemática)
Escalada
Haz
Voraz
A*
IDA*
SMA*
Completos, pero mucha memoria
Menos memoria, pero incompletos

- Métodos **iterativos aproximados**, diseñados para resolver problemas difíciles de optimización donde los métodos heurísticos no son efectivos

- Conceptualmente *por encima* de los métodos heurísticos

- Marco general para crear algoritmos híbridos combinando conceptos derivados de la IA, evolución biológica y mecanismos estadísticos

- Obtienen soluciones optimizadas a problemas complejos en tiempos razonables – **sin garantía** de optimalidad/completitud

Meta-heurísticas **constructivas**:

- Construyen iterativamente la solución en un espacio de estados (soluciones parciales) incorporando sucesivamente elementos a la misma

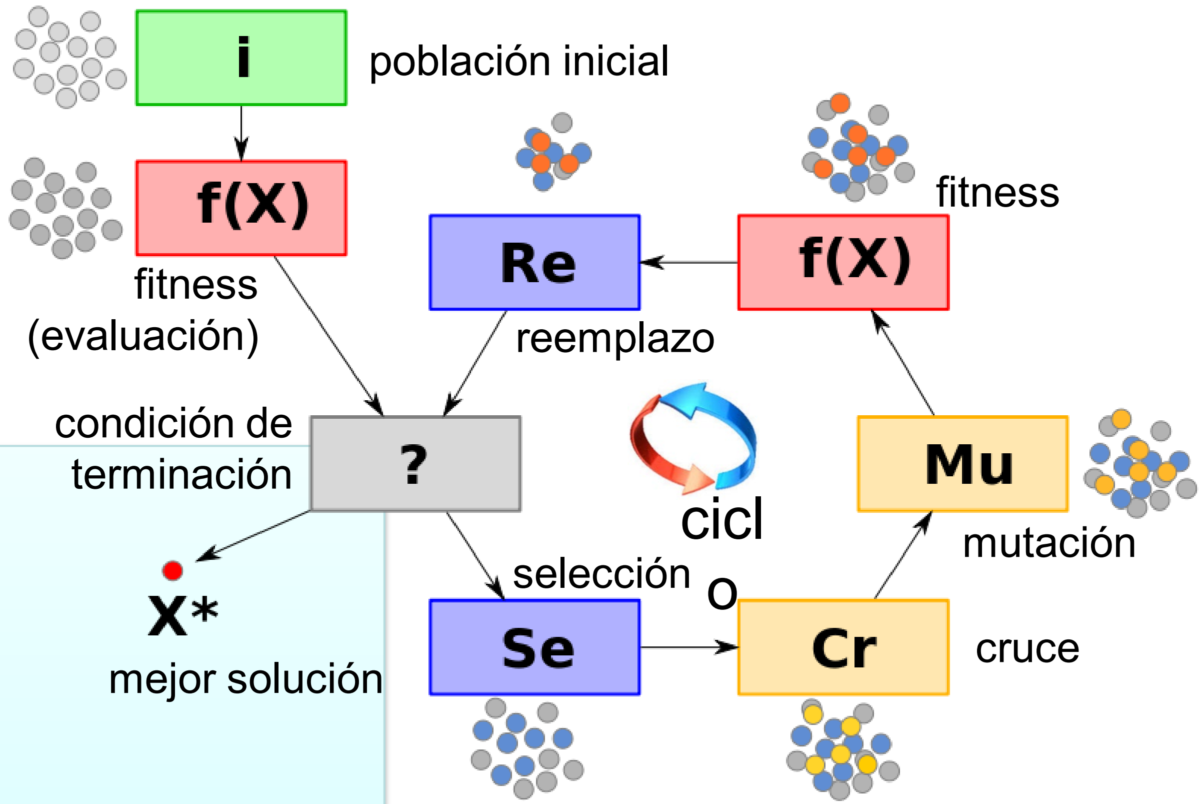
- Incorporan criterios para seleccionar los elementos de una buena solución

- Ej. GRASP (Greedy Randomized Adaptative Search Procedure, 1ª fase)

Meta-heurísticas **de mejora**:

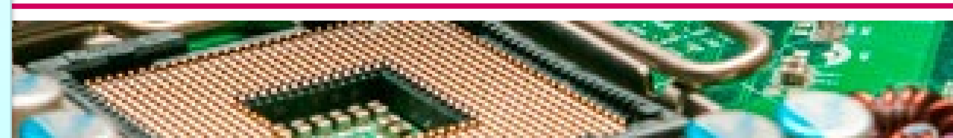
- Transforma el espacio de soluciones de partida. **Mejora iterativa** (pérdida de optimalidad e incompletos)
- Suelen incorporar estrategias para escapar de óptimos locales:
  - Arranque múltiple, para reiniciar la búsqueda (multi-arranque y backjumping)
  - Entorno variable, para modificar la estructura (búsqueda Tabú con memoria y búsqueda en haz)
  - Búsqueda no monótona, con modificaciones que pueden empeorar (enfriamiento simulado)

Meta-heurísticas **de mejora**. Ejemplo **Algoritmo Genético**



https://commons.wikimedia.org/wiki/File:Evolutionary_algorithm.svg

```
t ← 0;    terminar ← False;
inicializar P(0);
while not(terminar)
   S ← Selección [P(t)];
   Q ← Cruce [S];
   Q' ← Mutación [Q];
   P(t+1) ← Reemplazo [P(t), S, Q'];
   t ← t+1
   terminar ← (Convergencia [P(t+1)]) OR (t>límite)
end_while
```



Ejemplo. Algoritmos Genéticos en logística

Optimización de diseño de rutas de vehículos usando algoritmos genéticos

Aplicación de Algoritmos genéticos, una estrategia para disminuir costos de operación en la planificación de rutas

Aplicación de algoritmos genéticos para la programación de tareas en una celda de manufactura

Programación de las actividades logísticas con algoritmos genéticos

Uso de algoritmos genéticos para resolver el modelo determinista y estocástico para el diseño de una red de recogida de residuos

Genetic Algorithm for Logistics Scheduling Problem

Genetic Algorithm based Logistics Route Planning

Network model and optimization of reverse logistics by hybrid genetic algorithm

Application of Improved Genetic Algorithm in Logistics Transportation

ALGORITMOS INTELIGENTES PARA PROBLEMAS LOGÍSTICOS

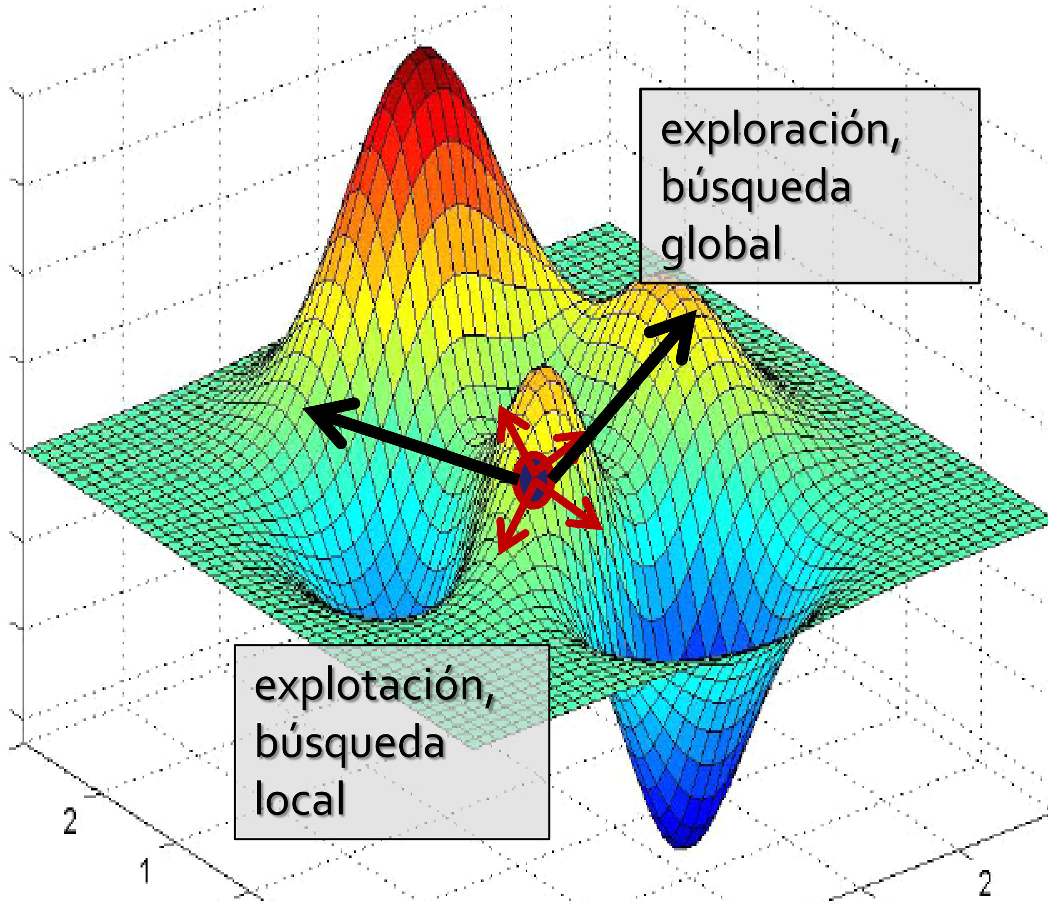
RouteME
La elección inteligente en logística

Exploración de alternativas en el espacio de soluciones.

- Busca alternativas un amplio espacio de estados ≈ búsqueda global
- Evita convergencia prematura

Explotación de alternativas vecinas de una buena solución

- Se focaliza sobre el entorno de buenas soluciones (alto fitness) ≈ búsqueda local en la vecindad



Decisión determinista
Meta-heurísticas. Clasificación

Heurísticas de Trayectorias:

- Búsqueda tabú
- Búsqueda local guiada
- Métodos multi-arranque
- GRASP
- Enfriamiento simulado, etc.

Heurísticas poblacionales:

- Búsqueda dispersa
- Algoritmos evolutivos y genéticos
- Algoritmos meméticos
- Sistemas de hormigas
- Inteligencia de enjambre, etc.

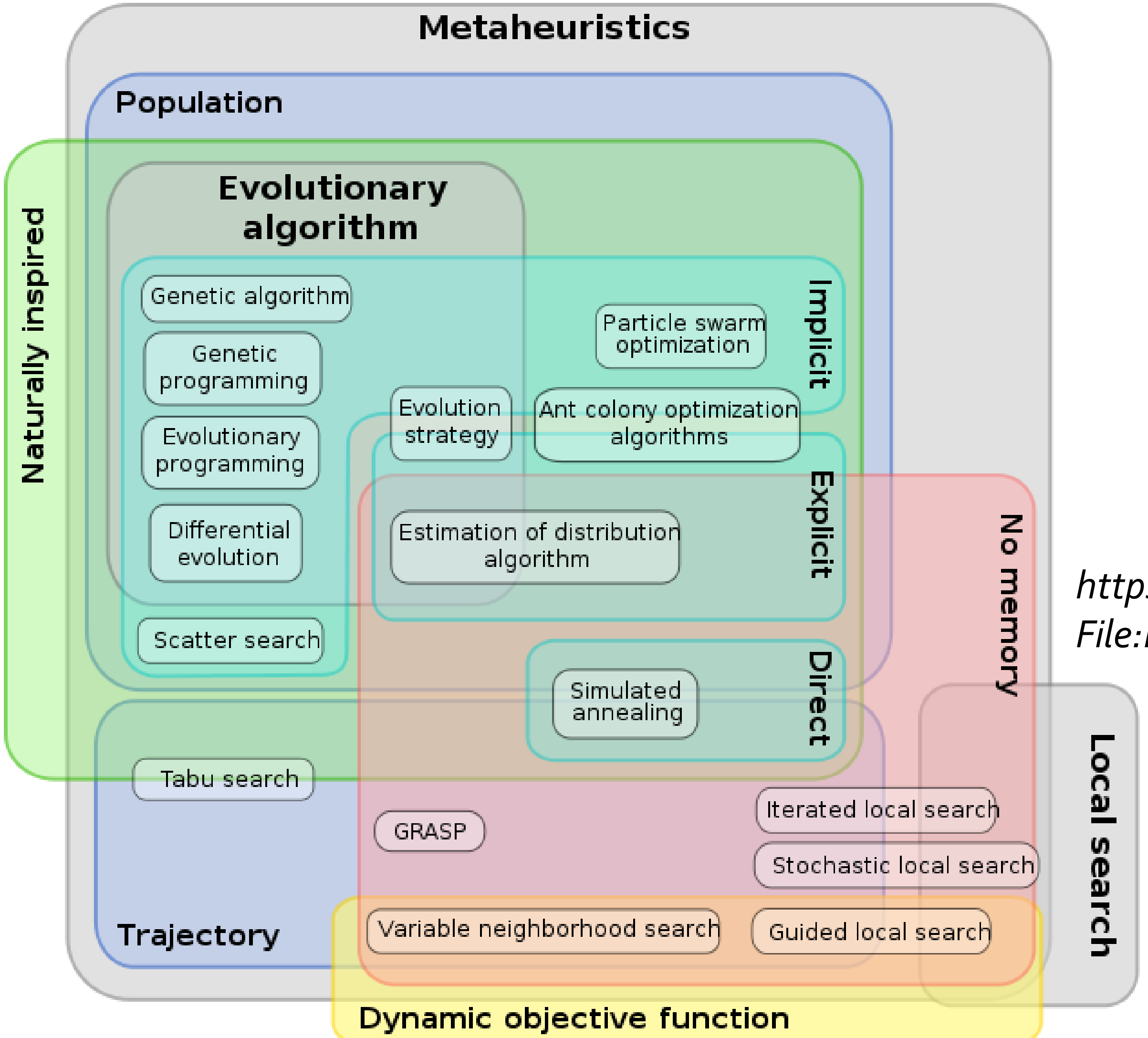
Inspiradas o no en la naturaleza:

- **Inspiradas**: Algoritmos Genéticos, Hormigas, Enfriamiento Simulado, etc.
- **No inspiradas**: Búsqueda dispersa, GRASP

Aleatorios *vs.* deterministas:

- **Aleatorias**: Algoritmos Genéticos, evolutivos, etc.
- **Deterministas**: Tabú, Búsqueda Dispersa, etc.

…



https://commons.wikimedia.org/wiki/File:Metaheuristics_classification.svg

Entornos para **trabajo con meta-heurísticas**

Opt4J. Entorno libre

A Modular Framework for Meta-heuristic Optimization

Disponible (ejecutable y fuente) en: https://sdarg.github.io/opt4j/

Formulación sencilla de problemas utilizando librerías implementadas en Java

HeuristicLab http://dev.heuristiclab.com/

Entorno generalista. C#.

PARADISEO https://nojhan.github.io/paradiseo/

API de programación en C++

jMetal https://jmetal.sourceforge.net/

Framework de optimización multi-objetivo en Java

Entornos para **trabajo con meta-heurísticas**

- Multi-lenguaje: PISA
- Java: ECJ 23
- C++: ECF, GAlib , Open BEAGLE
- Python: Distributed Evolutionary Algorithms , Pyevolve

- Toolbox **libres** sobre AG: GPLAB, GA Toolbox, GA_framework:

- Toolbox **comerciales**: MATLAB & Global Optimization Toolbox

MathWorks®

Algoritmo de las Hormigas

Algoritmos Genéticos

GRASP

Enfriamiento Simulado

Heurística h1(n)

Búsqueda Tabú

Algoritmo A*

Heurística h2(n)

Algoritmo de las Abejas

Heurística h3(n)

¿Cuál es mejor?

¿Qué ocurre cuando tenemos una métrica multi-objetivo?

a) **Combinación de objetivos**, con posibles pesos w_i (se convierte en problema mono-objetivo):

$$\text{Min } \Sigma_{i=1,m}\ w_i\ f_i(X)$$

Ejemplo (viajante multiobjetivo):

$$\text{Min } (w_1 \text{ Coste } (x_1, x_2, \ldots, x_n) + w_2 \text{ Distancia } (x_1, x_2, \ldots, x_n) + w_3 \text{ Tiempo } (x_1, x_2, \ldots, x_n))$$

Ventajas: Sencillez, Conversión a problema mono-objetivo

Inconvenientes: Diferencia en valor de los objetivos, no realiza una optimización multiobjetivo.

Devuelve: Una única solución

Variante: Optimización Mediante Metas

Se define un valor-meta de cada objetivo y se minimizan las desviaciones respecto a las metas definidas en cada objetivo: Min $\Sigma_{i=1,m}\ w_i\ [f_i(X^*) - f_i(X)]$

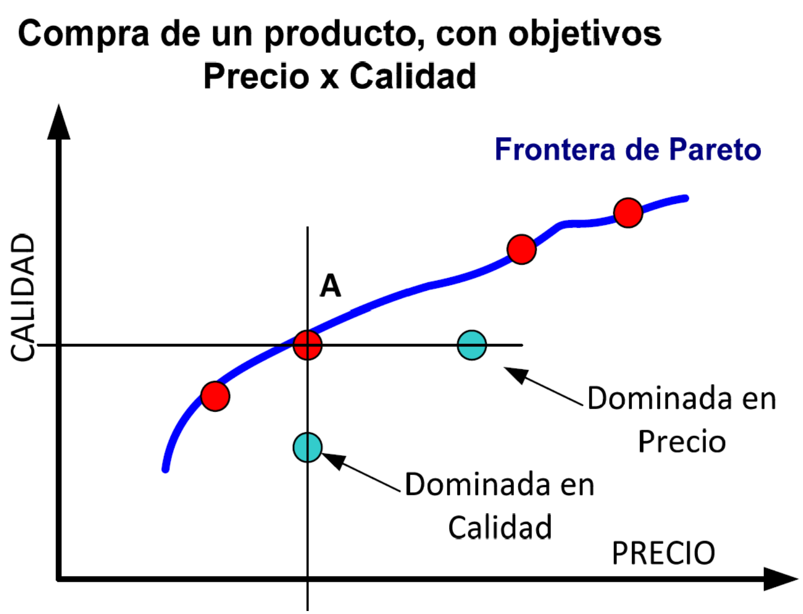
¿Qué ocurre cuando tenemos una métrica multi-objetivo?

b) Métodos basados en la Eficiencia de Pareto

Ventajas: Trata realmente un problema multi-objetivo

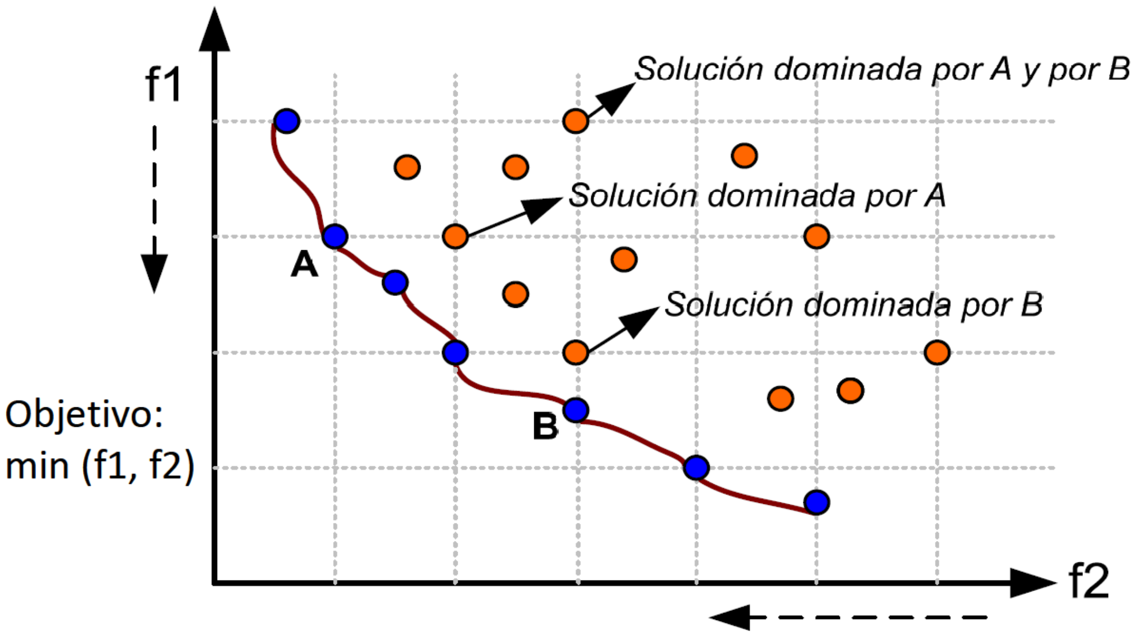
Inconvenientes: Mayor complejidad.

Devuelve: Conjunto de soluciones (frontera de Pareto, o puntos cercanos a ella)



La mejora de un objetivo habitualmente empeora otro

¿Qué ocurre cuando tenemos una métrica multi-objetivo?

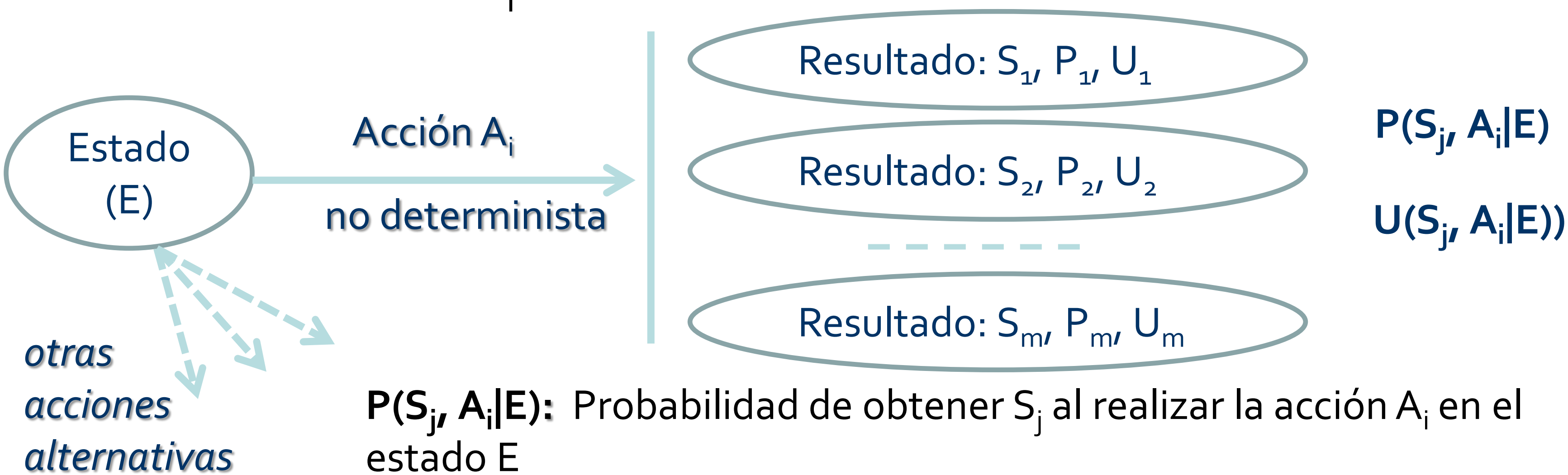


- Un resolvedor multiobjetivo ofrecerá un Conjunto de Soluciones de la Frontera de Pareto *(o soluciones más cercanas al frente)*.
- El frente puede tener una, muchas, o incluso infinitas.
- Se prefiere que las soluciones estén bien distribuidas.

Decisión determinista *vs.* no determinista

¿Qué pasa cuando las decisiones no son deterministas?
Es decir, cuando **no sabemos** las consecuencias de nuestras decisiones

Se **conocen** las utilidades y probabilidades de las consecuencias de cada decisión A_i



$P(S_j, A_i|E)$: Probabilidad de obtener S_j al realizar la acción A_i en el estado E

$U(S_j, A_i|E))$: Utilidad del estado S_j al realizar la acción A_i en el estado E

Utilidad Esperada de realizar la acción A_i en el estado E:

$$UE(A_i|E) = \Sigma_j [P(S_j, A_i|E) * U(S_j, A_j|E)]$$

Principio de **Máxima Utilidad Esperada**

un agente racional debe elegir aquella Acción (A_i) que maximice la Utilidad Esperada (o minimice la penalización)

¿Pero los seres humanos **somos siempre racionales**?

En un concurso he ganado un millón de euros. Puedo aceptar el premio o jugármelo con una moneda. Si sale cara, pierdo todo. Si sale cruz, gano tres millones de euros. ¿Qué hago?

Ejercicio

Ejemplo. Decisión con riesgo ante el lanzamiento de un nuevo producto de nuestra empresa

Ante el lanzamiento de un nuevo producto, se debe determinar la cantidad a producir. La demanda inmediata no se conoce, pero se estiman cuatro posibilidades V:{100,200, 300, 400}, con sus probabilidades P:{10,30, 40, 20} %.

Cada producto tiene un coste de fabricación F_i de 50€.

Si se vende inmediatamente, su precio de venta V_i será de 65 €, si no hay que ofertarlo a 40€.

¿**Cuántos productos** se deberían producir: 100, 200, 300, 400?

Ejemplo. Decisión con riesgo ante el lanzamiento de un nuevo producto de nuestra empresa

El beneficio en cada $caso_i$ es: $[65 V_i + 40 (F_i - V_i)] - 50F_i$

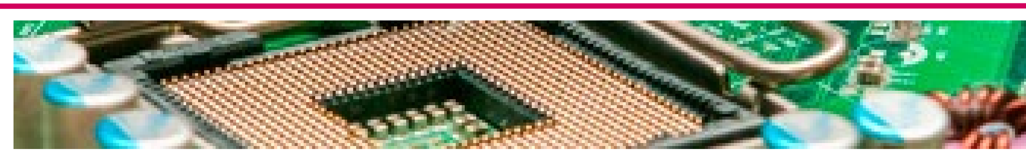
Ejercicio

No se conocen las probabilidades de la consecuencia de cada decisión A_i

Hay **distintos criterios** que combinan la utilidad y el riesgo:

- Permiten tomar decisiones más/menos arriesgadas
- Aplican criterios optimistas, pesimistas, equilibrados, agresivos, conservadores, etc.

No hay claramente un mejor criterio. Se suelen aplicar varios y quedarnos con el criterio que salga ganadora el mayor número de veces



Ejemplo. Decisión con incertidumbre para decidir la campaña publicitaria ante el lanzamiento de un nuevo producto

La campaña puede provocar una alta, baja o media demanda, pero se desconoce la probabilidad de cada consecuencia

Los beneficios esperados (utilidad), combinando impacto y coste de cada medio, serían:

	Estados Alcanzables			
¿Acciones?	Alta Demanda	Media Demanda	Baja Demanda	Con 1/3 probabilidad (igual verosimilitud, Laplace)
A1: ¿Prensa?	90	35	25	90/3 + 35/3 + 25/3 = 50
A2: ¿Radio?	100	40	20	100/3 + 40/3 + 20/3= 53.3
A3: ¿TV?	80	20	5	80/3 + 20/3 + 5/3 = 35

Ejemplo. Decisión con incertidumbre para decidir la campaña publicitaria ante el lanzamiento de un nuevo producto

Criterio optimista (Maximax, Hurwicz). Elige la alternativa que obtiene el máximo beneficio en su **mejor** estado alcanzable

	Alta Demanda	Media Demanda	Baja Demanda	Mejor
¿Prensa?	90	35	25	90
¿Radio?	100	10	20	100
¿TV?	80	20	5	80

Ejemplo. Decisión con incertidumbre para decidir la campaña publicitaria ante el lanzamiento de un nuevo producto

Criterio pesimista (Wald). Elige la alternativa que obtiene el máximo beneficio en su **peor** estado alcanzable

	Alta Demanda	Media Demanda	Baja Demanda	Peor
¿Prensa?	90	35	25	25
¿Radio?	100	10	20	10
¿TV?	80	20	5	5

Ejemplo. Decisión con incertidumbre para decidir la campaña publicitaria ante el lanzamiento de un nuevo producto

Criterio balanceado. Combinación de criterio optimista (factor p) y pesimista (1-p), donde $0<p<1$ se define de antemano

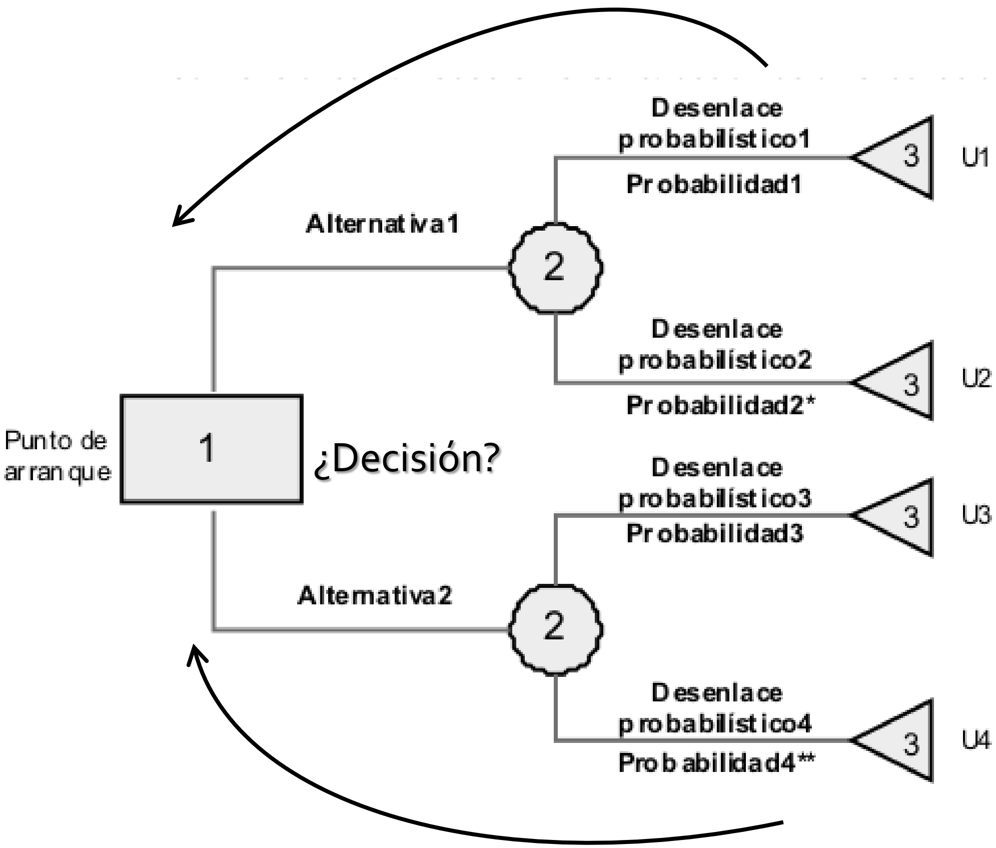
p=0.6	Alta Demanda	Media Demanda	Baja Demanda	Resultado Mejor*p + Peor * (1-p)
¿Prensa?	90*0.6	35	25*0.4	64
¿Radio?	100*0.6	10*0.4	20	64
¿TV?	80*0.6	20	5*0.4	50

Ejemplo. Decisión con incertidumbre para decidir la campaña publicitaria ante el lanzamiento de un nuevo producto

Criterio balanceado. Combinación de criterio optimista (factor p) y pesimista (1-p), donde 0<p<1 se define de antemano

p=0.3	Alta Demanda	Media Demanda	Baja Demanda	Resultado Mejor*p + Peor * (1-p)
¿Prensa?	90*0.3	35	25*0.7	44.5
¿Radio?	100*0.3	10*0.7	20	37
¿TV?	80*0.3	20	5*0.7	27.5

- **Nodos de azar (círculos):** representan puntos en los que la naturaleza elige un estado. Cada rama puede tener asociada o no una probabilidad
- **Nodo de decisión (cuadrados):** representan puntos de decisión, con tantos arcos salientes como alternativas existan
- **Nodo terminal u hoja (triángulos):** nodos finales de cada rama, que representa el estado alcanzable tras la sucesión de decisiones. Es evaluable mediante la función de utilidad

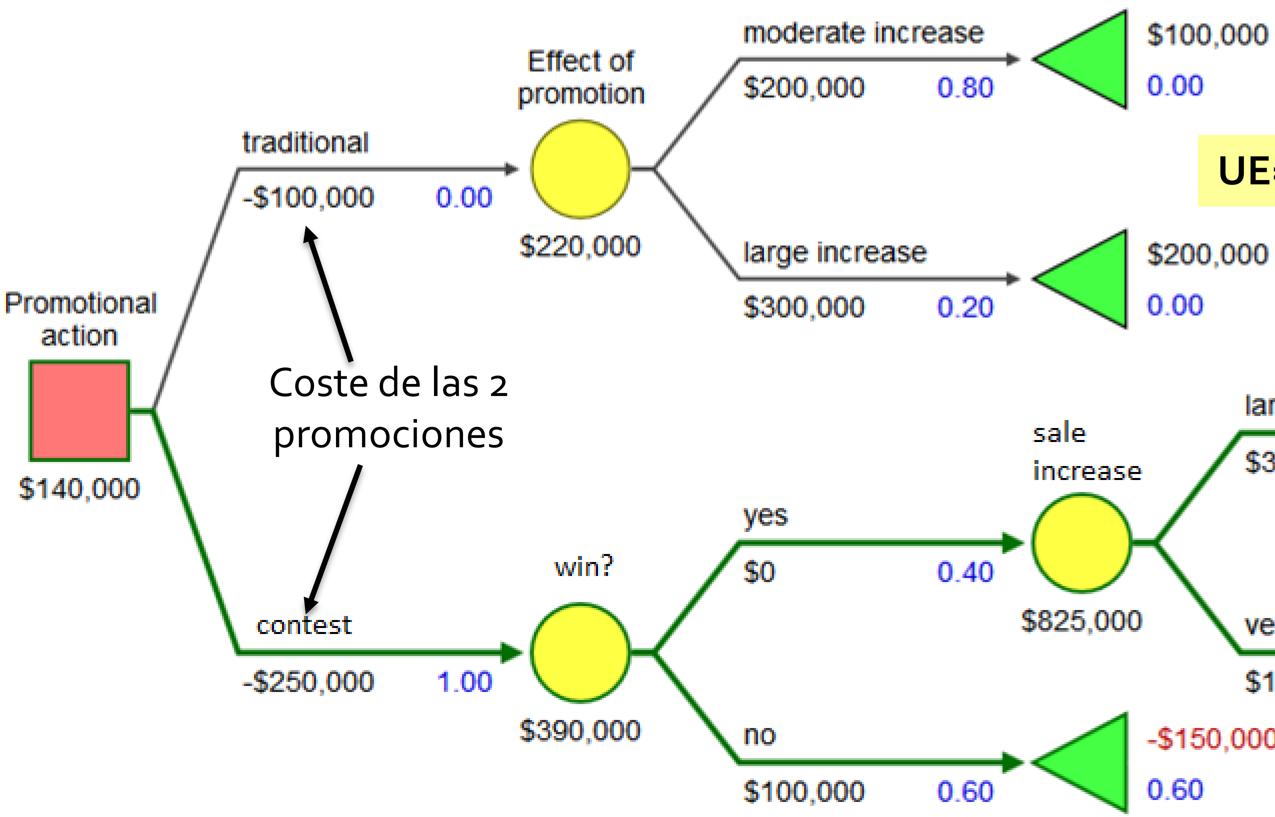


Ejemplo. Estrategias promocionales para aumentar las ventas de nuestra empresa

- Medios conservadores (prensa, folletos, etc.), con un coste de 100.000\$, estimando que el incremento de ventas aportará 200.000\$ adicionales con probabilidad del 80% y 300.000\$ con 20%

- Presentarse a un concurso de popularidad en TV. El coste se estima en 250.000\$, con posibilidad de éxito del 40%. Si no se gana el concurso, el incremento de ventas supondrá solo 100.000\$ adicionales. Si se gana, el incremento podría aportar 300.000\$ adicionales, con probabilidad del 25%, o 1.000.000\$ adicionales con probabilidad del 75%

Ejemplo. Estrategias promocionales para aumentar las ventas de nuestra empresa



UE= 100k*0.8 + 200 k* 0.2= 120k

UE= (50k * 0.25 + 750k * 0.75) * 0.4 - 150k*0.6 = 140k

Toma de decisiones con incertidumbre, donde la incertidumbre se debe a las decisiones de otros oponentes o al propio azar

Se define una **matriz de peso/pago/ganancias** que identifica el beneficio ante todas las posibles decisiones. Ej típico de teoría de juegos: dilema del prisionero. ¿Qué haría yo?

Matriz de Pagos	***Acción:* Yo delato**	***Acción:* Yo no delato**
***Estado:* Cómplice delata**	**Ambos condenados a 5 años**	**Yo condenado a 10 años, cómplice libre**
***Estado:* Cómplice no delata**	**Yo libre, cómplice a 10 años**	**Ambos condenados a 1 año**

Estrategias dominantes

Cuando cada oponente/jugador tiene una estrategia dominante, la combinación de todas se llama equilibrio de estrategias dominantes

- Una estrategia e para un jugador domina a una estrategia e', si Utilidad (e) ≥ Utilidad (e') para toda estrategia posible
- **Perfil de equilibrio**: sea cual sea la decisión del otro, a ambos les conviene elegir dicha estrategia (es **lo racional**, aunque puede que no sea lo óptimo)
- Si cada oponente tiene una estrategia dominante se puede predecir el resultado de las decisiones

Ejemplo. Decidir o no una campaña publicitaria en función de la competencia

Dos empresas A y B venden productos rivales y tienen que decidir si emprenden o no una campaña publicitaria. La decisión que tome cada una afectará a las ventas propias y de la otra empresa. Cada empresa desea obtener su máximo beneficio

	B: Si publicidad	B: No Publicidad
A: Si publicidad	(A: 10, B: 5)	(A: 15, B: 0)
A: No Publicidad	(A: 6, B: 8)	(A: 20, B: 2)

Ejemplo. Decidir o no una campaña publicitaria en función de la competencia. ¿Existe una estrategia dominante en A o B?

	B: Si publicidad	**B: No Publicidad**
A: Si publicidad	(A: 10, B: 5)	(A: 15, B: 0)
A: No Publicidad	(A: 6, B: 8)	(A: 20, B: 2)

Ejercicio

No siempre hay estrategias dominantes, por lo que se buscan otras estrategias

Equilibrio de Nash

Dado un juego con N jugadores, las estrategias $\{S^*_1, S^*_2, \ldots\ldots, S^*_n\}$ son un Equilibrio de Nash si, para cualquier $jugador_i$, la estrategia S^*_i es la mejor respuesta a las estrategias $\{S^*_1, S^*_2, \ldots, S^*_{i-1}, S^*_{i+1}, \ldots, S^*_n\}$ de los demás jugadores:

$$U(S^*_1, S^*_2, \ldots, S^*_{i-1}, S^*_i, S^*_{i+1}, \ldots, S^*_n) \geq U(S^*_1, S^*_2, \ldots, S^*_{i-1}, S_i, S^*_{i+1}, \ldots, S^*_n), \ \forall Jugador_i \in N, \ \forall S_i \in S$$

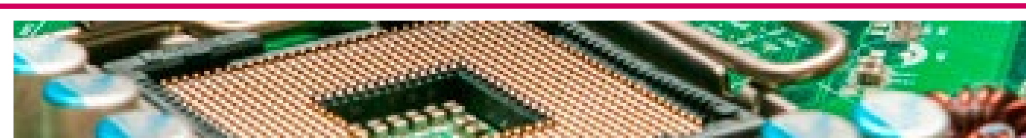
Dados dos jugadores A y B, las estrategias (a*, b*) forman un Equilibrio de Nash, si la estrategia a* es óptima para A frente a la estrategia b* (de B), y b* es óptima para B frente a la estrategia a* (de A)

No siempre hay un equilibrio de Nash

Hay situaciones con **estrategias ganadoras**, que garantizan siempre la mejor decisión (haga lo que haga el contrario) – aunque a veces son muy caras de determinar. Ej. Juego de Nim

Existen **estrategias reactivas**, que se basan en lo que históricamente va realizando el oponente. Ej. en dilema de prisioneros:

- Siempre delatar o siempre no-delatar
- Ojo por ojo (hacer lo que te han hecho)
- Gallina (lo contrario), o del torito (traicionarlo), etc.



Conclusiones

Existen muchas **técnicas informáticas** que se pueden aplicar en la logística

- Para obtener soluciones o buenas soluciones (óptimas)
- Garantizar la optimalidad es un proceso muy caro → técnicas heurísticas, meta-heurísticas y de IA

- Se pueden utilizar para tomar decisiones, aunque existen decisiones cuyas consecuencias son desconocidas
  - Las probabilidades no se conocen (incertidumbre)
  - Existen oponentes que añaden nueva incertidumbre