Lógica de primer orden, unificación y resolución.
Definición de estado y transición.
Planificación en el mundo de los cubos.

Brenda Aymerich, Oscar Esquivel, César Mata y Andrés Solano

Lógica de primer orden

Breve historia

Origen

Los primeros sistemas lógicos se remontan a Aristóteles, sin embargo, la lógica de primer orden, tal y como la conocemos, fue originalmente propuesta por Gottlob Frege (*Conceptografía*) en la segunda mitad del siglo XIX. Posteriormente publicaría un trabajo titulado *Los fundamentos de la aritmética* en el cual refinaría sus ideas.

No obstante, no fue hasta que Giuseppe Peano y Bertrand Russell modificaron la notación de Frege para que las ideas de este alcanzaran una mayor audiencia.

En la década de los años treinta, Kurt Göedel y Jacques Herbrand estudiaron la noción de lo computable basada en derivaciones. Su trabajo puede verse como el origen de la computación como deducción.

Origen

Además, Herbrand discutió un conjunto de reglas para manipular ecuaciones algebraicas en términos que pueden verse ahora como un bosquejo de la unificación.

Treinta años más tarde, Robinson publicó su artículo sobre la demostración automática. En este trabajo se introduce el principio de resolución, la noción de unificación y un algoritmo para su cómputo.

Durante los años siguientes la lógica de primer orden ha recibido aportes como demostraciones y teoremas de personajes importantes como Göedel y Turing, lo cual le ha permitido establecerse como un sistema formal sólido dentro de la metalógica.

Lógica de primer orden

Definición y conceptos generales

Lógica de predicados

Un predicado es lo que se afirma de un sujeto. En lógica, también se conoce como predicado a las relaciones entre sujetos.

La Tierra *<u>es un planeta</u>*
(expresión) (predicado)

Júpiter *<u>posee más masa que</u>* *la Tierra*
(expresión) (relación) (expresión)

Lógica de predicados

Los predicados pueden ser unarios, binarios, ternarios, dependiendo de la cantidad de argumentos.

En nuestro caso, la lógica de primer orden nos permite representar conceptos que contienen relaciones entre objetos simples:

Padre e hijo

Sin embargo, la lógica de predicados es limitada. No es posible representar relaciones entre relaciones o propiedades de relaciones al mismo tiempo:

Padre es una relación familiar

Objetivo principal

Describir los objetos que conforman un universo de discurso, así como las relaciones que existen entre ellos con el para obtener conclusiones nuevas a partir de lo anterior.

Para entender cómo se logra, es necesario analizar la lógica de primer orden como un sistema formal considerando:

Sintaxis: definir cuáles expresiones pertenecen a este conjunto.

Semántica: determinar qué hace verdadera o falsa una cláusula

Reglas de razonamiento: obtener nuevas conjeturas.

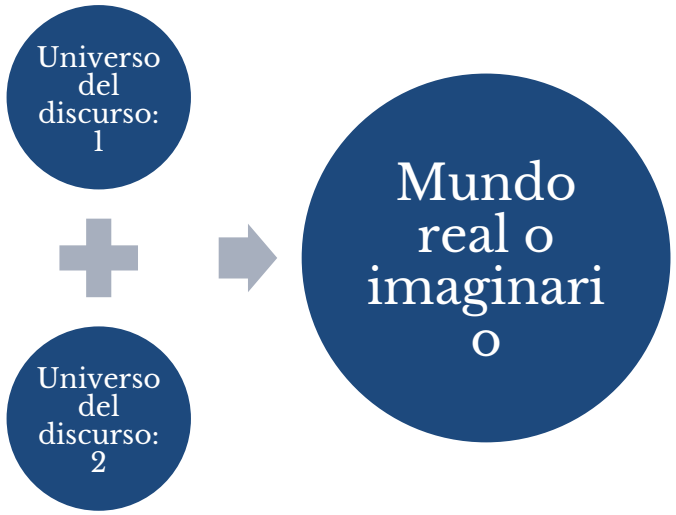
Lógica de primer orden

Conceptos del lenguaje

Lenguaje

El lenguaje de la lógica de primer orden está representado por un *conjunto de símbolos* que nos permite expresarnos sobre los objetos de un determinado dominio.

El conjunto de todos esos objetos se conoce como *universo de discurso*. Un *objeto* es algo sobre lo cual queremos expresarnos sin importar su naturaleza.



Funciones

Una función es un tipo especial de relación entre los objetos del dominio de discurso que mapea un conjunto de objetos de entrada a un objeto único de salida.

El conjunto de todas las funciones del universo del discurso se conoce como *base funcional*.

Ejemplo de función: 'madre de'

Conjunto de objetos de entrada

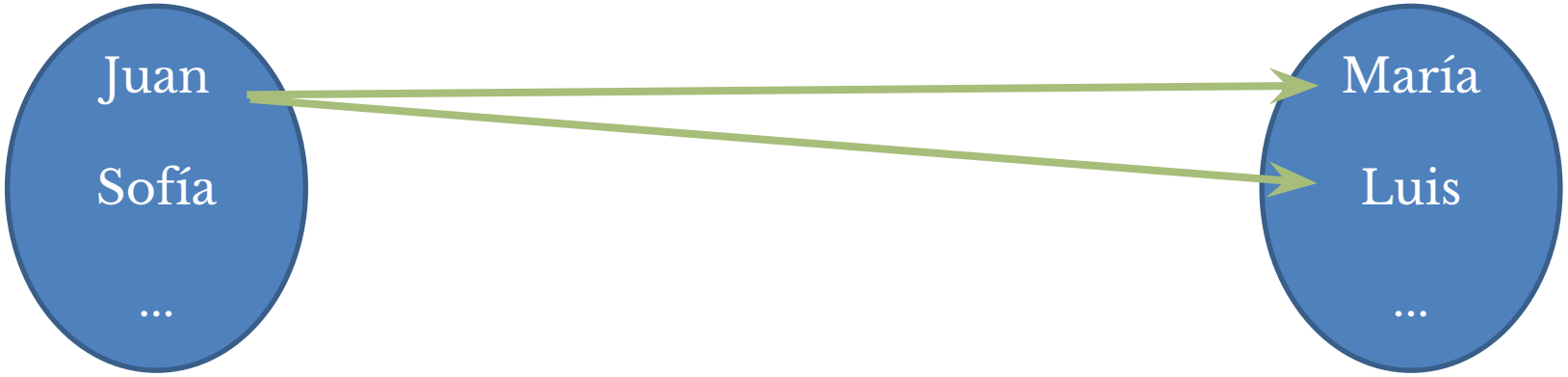
Conjunto de objetos de salida

Predicados

Un predicado puede utilizarse para definir hechos o verdades aceptadas dentro de un universo de discurso.

El conjunto de todos los predicados usados en la conceptuación se conoce como *base relacional*.

Ejemplo de predicado: 'hijo de'



Conjunto de objetos de entrada

Conjunto de objetos de salida

Variables, constantes y cuantificadores

Las variables se utilizan para representar los objetos del universo de discurso y se representan normalmente por cualquier secuencia de caracteres que inicie con una letra mayúscula.

Las constantes son representaciones que referencian siempre la misma entidad. Podría verse como una función que recibe 0 argumentos.

Cuantificadores:

Para todo ($\forall$): nos permite expresar hechos acerca de todos los objetos en el universo del discurso, sin enumerarlos.

Existe ($\exists$): nos permite expresar la existencia de un objeto en el universo de discurso con cierta propiedad en particular.

Conectivas

La lógica de primer orden incorpora las conectivas de la lógica proposicional para combinar los predicados, constantes, variables y cuantificadores.

La Tierra y Marte poseen menos masa que Júpiter

Conectiva	**Lenguaje natural**	**Símbolo**
Negación	no	$\neg, \sim$
Conjunción	y	$\wedge, \ \& \ , \cdot$
Disyunción	o	$\vee$
Condicional	si... entonces	$\rightarrow, \supset$
Bicondicional	si y sólo sí	$\leftrightarrow, \equiv$
Disyunción excluyente	o bien... o bien	$\nleftrightarrow, \oplus, \not\equiv, W, \veebar$

Lógica de primer orden

Sintaxis

Definición formal de los términos

Un término de la lógica de primer orden puede definirse de la siguiente forma:

$$t ::= x \mid c \mid f(t, \ldots, t)$$

Donde x es una variable; c es una función tal que |c| = 0; y f una función tal que |f| > 0.

calif(estudiante(juan), rob)

Lógica de primer orden

Semántica

Definición intuitiva de la semántica

Para lograr una definición formal de la semántica es más sencillo ejemplificar una idea intuitiva para enlazar primero los conceptos clave que se han mencionado durante la presentación.

Ejemplo

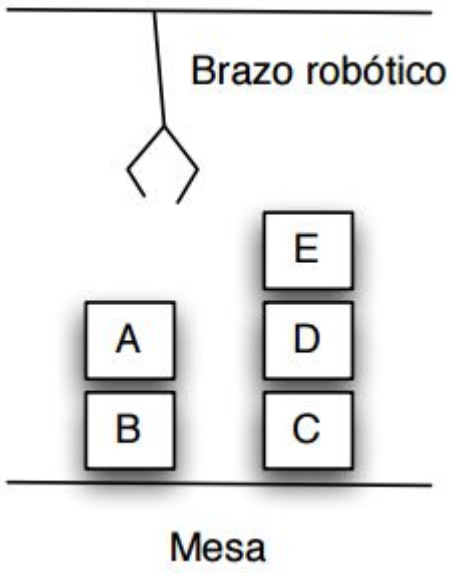
Problema: Representar que al menos un bloque no tiene otro encima.

Una posible solución sería:

$$\exists x\ Bloque(x) \wedge Libre(x)$$

El dominio puede especificarse en término de conjuntos:

$$D = \{Mesa, Brazo, A, B, C, D, E\}$$



Ejemplo

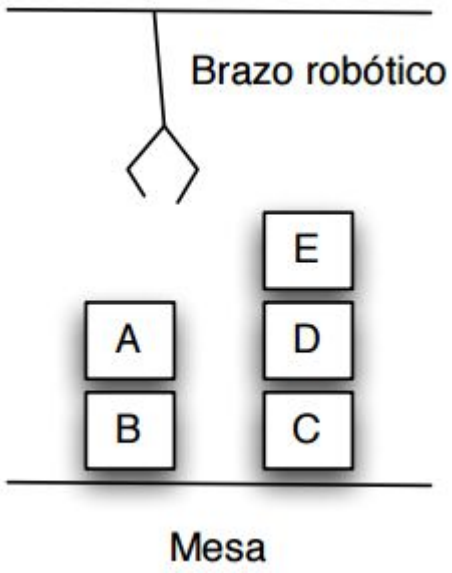
Podemos definir otro conjunto que reúna todos los bloques de nuestro dominio:

$$B = \{A, B, C, D, E\}$$

Con lo cual, la nueva interpretación sería:

$$\exists x\ Bloque(x) \wedge Libre(x) \equiv \forall x\ Libre(x)$$

(si el nuevo dominio es *B)*.



Explicación y definición

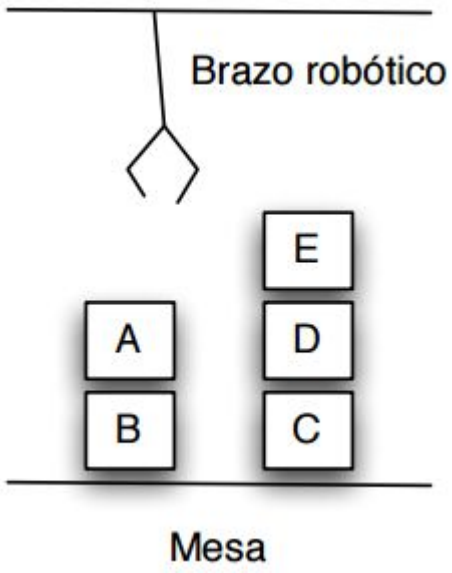
Para obtener una interpretación sobre un predicado se debe formar el par *(D, V)*, donde *D* es el universo y *V* una función, tal que para cualquier predicado con argumentos *n*, retorna las tuplas correspondientes a la interpretación de dicho predicado.

Esta función cumple con las siguientes propiedades:

> Si el parámetro de *V* es una constante, la interpretación deberá ser el mismo valor de la constante.
>
> Si el parámetro es un predicado con *n* argumentos, la interpretación será un subconjunto del universo *D*.

Análisis de la base relacional del ejemplo

$$
\begin{aligned}
A^V &= A \\
B^V &= B \\
C^V &= C \\
D^V &= D \\
E^V &= E \\
\mathrm{Sobre}^V &= \{(A,B),(E,D),(D,C)\} \\
\mathrm{EnLaMesa}^V &= \{B,C\} \\
\mathrm{Libre}^V &= \{A,E\} \\
\mathrm{PorEncima}^V &= \{(A,B),(E,D),(E,C),(D,C)\}
\end{aligned}
$$



Lógica de primer orden

Inferencia

Definición

El proceso de inferencia en la lógica de primer orden se puede ver como una manipulación de los enunciados, también llamados premisas, con el fin de obtener una conclusión nueva a partir de las anteriores.

Estas manipulaciones se pueden formalizar mediante reglas de inferencia. Para esto analizaremos la unificación y resolución.

Unificación - Definición

La unificación es un proceso que consiste en encontrar una asignación de variables que haga idénticas a las fórmulas que se desea unificar. Su resultado se expresa como un conjunto de pares substitución/variable para cada una de las variables asignadas (este conjunto recibe el nombre de substitución).

El valor de substitución para una variable puede ser cualquier término del lenguaje lógico utilizado (exceptuando términos con la misma variable).

Ejemplo

Por ejemplo, se pueden unificar las fórmulas:

padre(X, hermano(Y))
padre(juan, Z)

utilizando el unificador { juan / X, hermano(Y) / Z }.

También se podría utilizar la substitución

{ juan / X, hermano(pedro) / Z, pedro / Y },

pero aquí se introduce una asignación suplementaria que no es necesaria para unificar. Para evitar introducir sustituciones arbitrarias, se utiliza el unificador más general, es decir, el que minimiza las restricciones impuestas a los valores de las variables.

Ejemplo

Entonces, los términos t_1 y t_2 son unificables si tienen algún unificador σ tal que $t_1\sigma = t_2\sigma$. Y, t es una instancia común de t_1 y t_2 si existe una sustitución σ tal que $t_1\sigma = t_2\sigma = t$.

t_1	t_2	Unificador	Instancia común
$f(x, g(z))$	$f(g(y), x)$	$[x/g(z), y/z]$	$f(g(z), g(z))$
$f(x, g(z))$	$f(g(y), x)$	$[x/g(y), z/y]$	$f(g(y), g(y))$
$f(x, g(z))$	$f(g(y), x)$	$[x/g(a), y/a]$	$f(g(a), g(a))$
$f(x, y)$	$f(y, x)$	$[x/a, y/a]$	$f(a, a)$
$f(x, y)$	$f(y, x)$	$[y/x]$	$f(x, x)$
$f(x, y)$	$g(a, b)$	No tiene	No tiene
$f(x, x)$	$f(a, b)$	No tiene	No tiene
$f(x)$	$f(g(x))$	No tiene	No tiene

Composición de sustituciones

Al realizar una composición de substituciones σ_1 y σ_2 se obtiene la substitución $\sigma_1\sigma_2$ definida como $x(\sigma_1, \sigma_2) = (x\sigma_1)\,\sigma_2$ para toda variable x.

Por ejemplo: Si $\sigma_1 = [x / f(z, a), y / w]$ y $\sigma_2 = [x / b, z / g(w)]$, entonces

- $x\sigma_1\sigma_2 = (x\sigma_1)\sigma_2 = f(z, a)\sigma_2 = f(z\sigma_2, a\sigma_2) = f(g(w), a)$

- $y\sigma_1\sigma_2 = (y\sigma_1)\sigma_2 = w\sigma_2 = w$

- $z\sigma_1\sigma_2 = (z\sigma_1)\sigma_2 = z\sigma_2 = g(w)$

- $w\sigma_1\sigma_2 = (w\sigma_1)\sigma_2 = w\sigma_2 = w$

- Por tanto, $\sigma_1\sigma_2 = [x / f(g(w), a), y / w, z / g(w)]$.

Resolución - Definición

La resolución es una técnica utilizada para probar teoremas en lógica. Utiliza refutación para comprobar una sentencia determinada.

Ésta se basa en negar la sentencia original para intentar crear una contradicción, demostrando de esta manera que la sentencia original es verdadera. Se aplica a sentencias escritas en forma clausulada.

Básicamente, se busca demostrar que la negación de una sentencia genera una contradicción con los hechos conocidos, es decir, que no es satisfacible.

Forma de trabajar de la resolución

(1) Se convierten todas las cláusulas a la forma clausal.

(2) Se niega el estatuto a probar, se convierte a forma clausal, y se agrega a las cláusulas del punto (1).

(3) Se repite lo siguiente hasta encontrar una contradicción.

Se escogen dos cláusulas que se llamarán cláusulas padre, luego, se busca en ellas un par de literales TI y ¬TI tal que una de ellas pertenezca a una cláusula padre y su negación a la otra cláusula padre. Se eliminan ambas literales y se crea el resolvente (se usa la unificación). La cláusula resultante se llama cláusula resolvente y si ésta es vacía, la contradicción fue encontrada; si no, la cláusula resolvente se agrega al conjunto de las demás.

Ejemplo - Problema

Se tiene el siguiente problema:

A. Jack es dueño de un perro

B. Quien es dueño de un perro es amante de los animales

C. Ningún amante de los animales mata a un animal

D. O Jack o Curiosidad mató al gato, cuyo nombre era Tuna

E. ¿Mató Curiosidad al gato?

Ejemplo - Pasos

Pasos de la Resolución:

(1) Expresar lo anterior como predicados de primer orden, por ejemplo:

Jack es dueño de un perro: $(\exists X)$ perro(X) $\wedge$ dueño(Jack, X).

(2) Se niega el estatuto a probar:

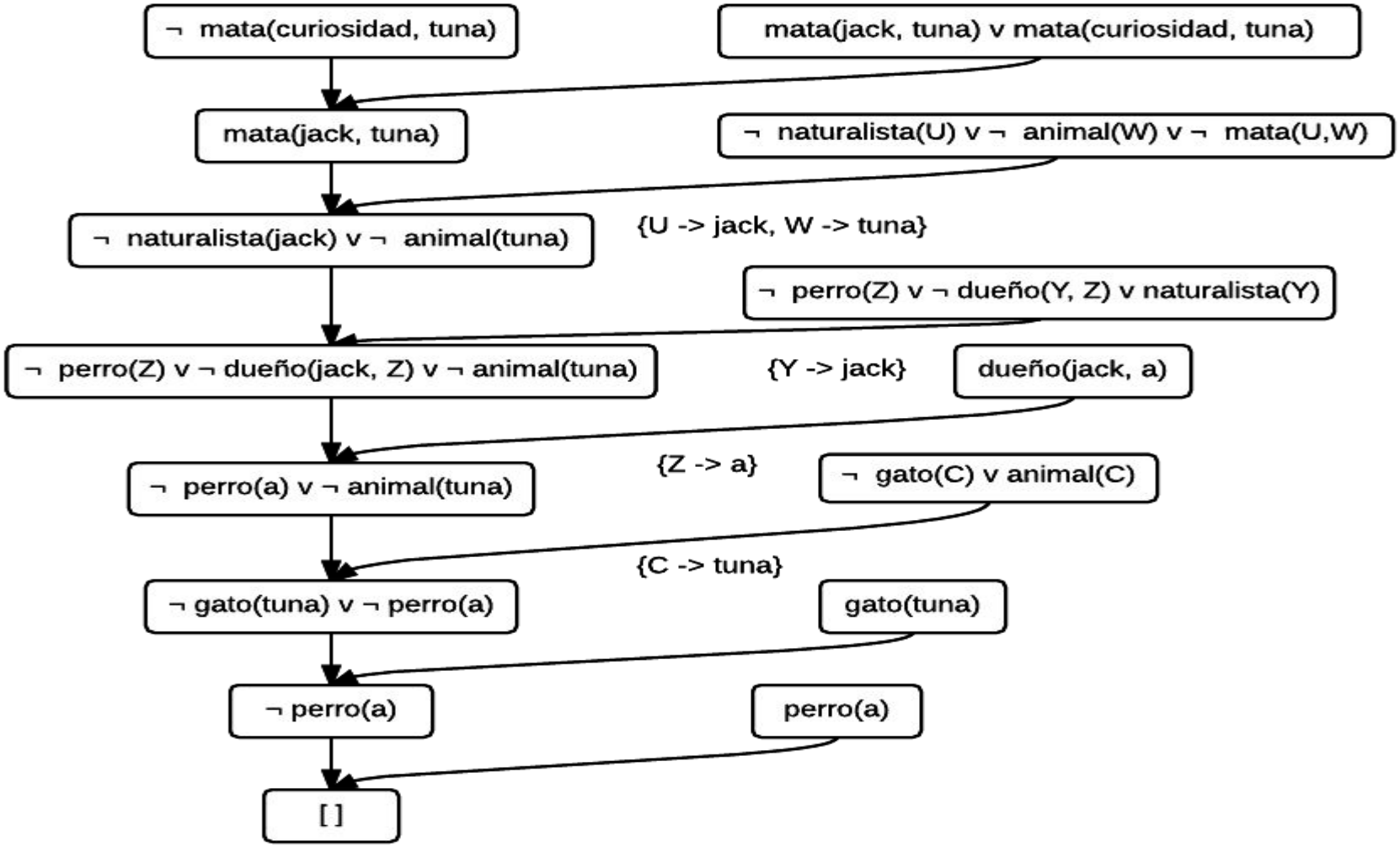
E. $\neg$ mata(curiosidad, tuna)

Ejemplo - Pasos

(3) Se siguen los 8 pasos para transformar las sentencias a su forma clausulada con lo que se obtiene el siguiente conjunto de cláusulas:

A1. perro(a)
A2. dueño(jack, a)
B. ¬ perro(Z) ∨ ¬ dueño(Y, Z) ∨ naturalista(Y)
C. ¬ naturalista(U) ∨ ¬ animal(W) ∨ ¬ mata(U,W)
D1. mata(Jack, tuna) ∨ mata(curiosidad, tuna)
D2. gato(tuna)
E. ¬ mata(curiosidad, tuna)
F. ¬ gato(C) ∨ animal

Ejemplo - Resolución



Definición de estado y transición

Planificación de problemas

- Se planifica la resolución de un problema con una conceptualización formal que simplifica todo el mundo del problema, lo abstrae, para que un robot pueda procesarlo.
- Se conceptualiza cada problema mediante “strips” formales.
- Se pueden agrupar estas strips para definir “estados”.
- Tomemos como ejemplo un robot que pueda movilizarse por el piso y estar en una de 4 distintas habitaciones, su posición entonces es dada por:

  “posiciónRobot(habitación)”

Estados

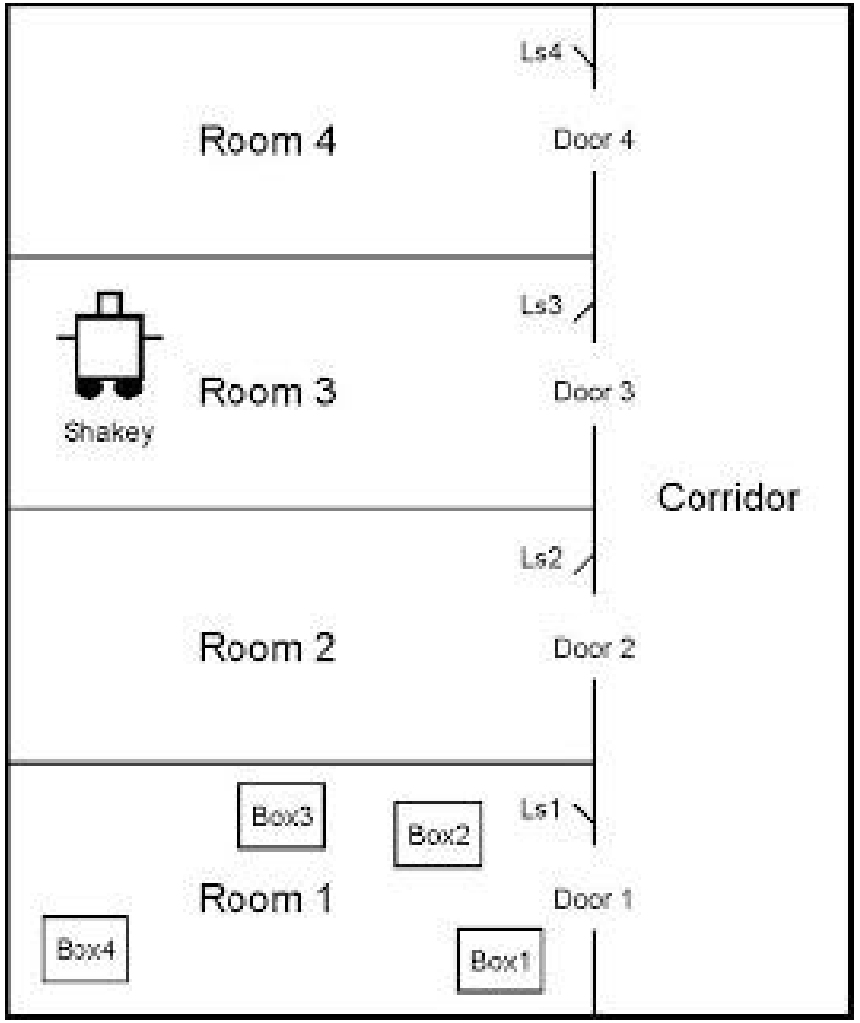
- Definidos por agrupaciones de strips.
- Definir un estado significa describir todas las condiciones del problema, describir todos los detalles relevantes.

Transiciones

- Los estados del problema pueden cambiar en el tiempo, a esto se le llama transición.
- Mediante una acción se puede cambiar las condiciones del problema, actualizando las strips que lo describen.
- Se avanza en la solución del problema, lo que permite que el robot analice el nuevo estado y pueda continuar resolviendo el problema según la lógica de primer orden

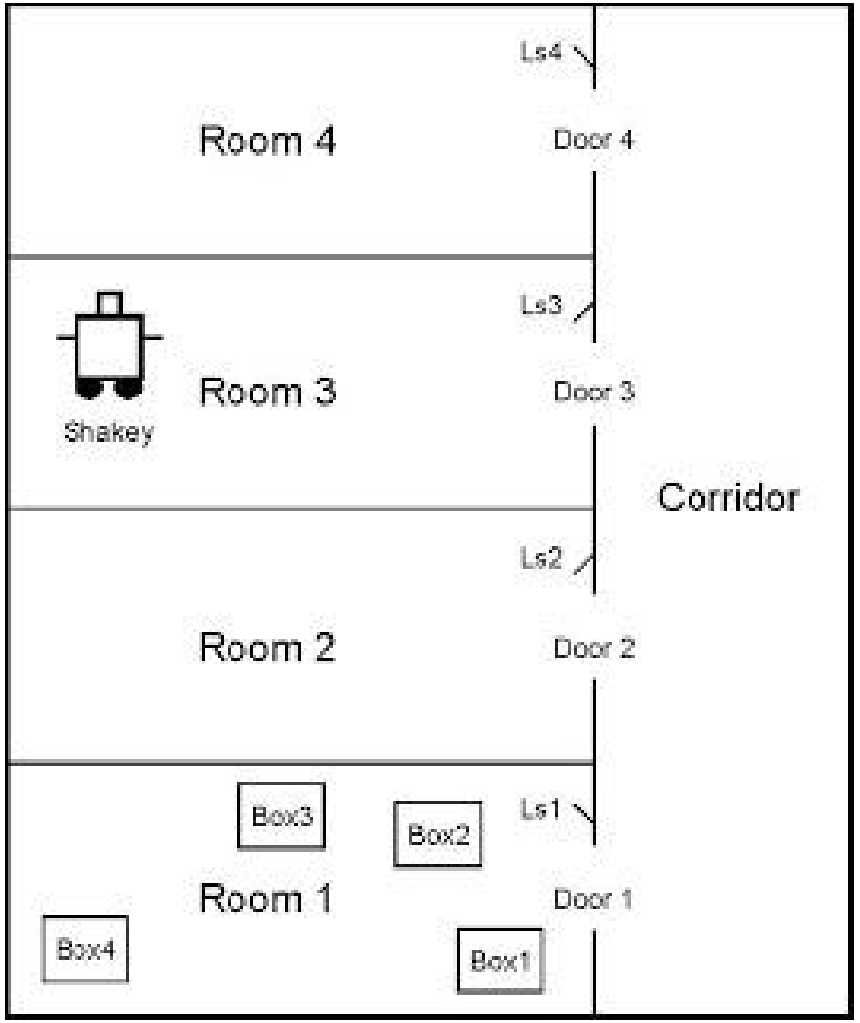
Ejemplo de “Shakey”

- Consiste en un robot que puede movilizarse entre habitaciones, transportar cajas, revisar que las luces estén encendidas o apagas en esas habitaciones, e incluso colocarse sobre las cajas para interactuar con los interruptores de las luces en las habitaciones.
- El estado inicial puede ser cualquiera, he ahí el problema, pero el estado final es siempre el mismo, ya que es la meta a la que queremos llegar y que al alcanzarla se considera que el problema fue resuelto.



Ejemplo de “Shakey”

- Estado final: “todas las luces apagadas y todas las cajas en la habitación 1”, se expresaría de la siguiente forma:
  - luzApagada(habitacion1)
  - luzApagada(habitacion2)
  - luzApagada(habitacion3)
  - luzApagada(habitacion4)
  - ¬ existeCajaHabitacion(habitacion2)
  - ¬ existeCajaHabitacion(habitacion3)
  - ¬ existeCajaHabitacion(habitacion4)
  - ShakeySinCaja



Ejemplo de “Shakey”

- Ejemplo de transición: entre la habitación 1 con la luz encendida y la misma habitación con la luz apagada. Para efectuar esta transición “Shakey” debe ejecutar la siguiente acción:
  - Descripción de la acción: apagarLuz(Habitacion1)

  - Precondiciones: Debe existir una caja en esa habitación, y la luz no debe estar apagada:
    - existeCajaHabitacion(Habitacion1) ^ ¬luzApagada (Habitacion1)

  - Efectos: Shakey no sostiene la caja, el estado de la luz es apagada
    - ShakeySinCaja ^ luzApagada(Habitacion1)

Estados y transiciones

- No sólo deben definirse las strips necesarias para definir los estados, sino que también deben definirse las acciones posibles.
- En la siguiente sección se explora a fondo otro ejemplo de mayor importancia retomando los conceptos de estados y transiciones, que muestra cómo se ponen estos conceptos en práctica para resolver los problemas que simplificamos con la planificación.

Planificación en el mundo de los cubos

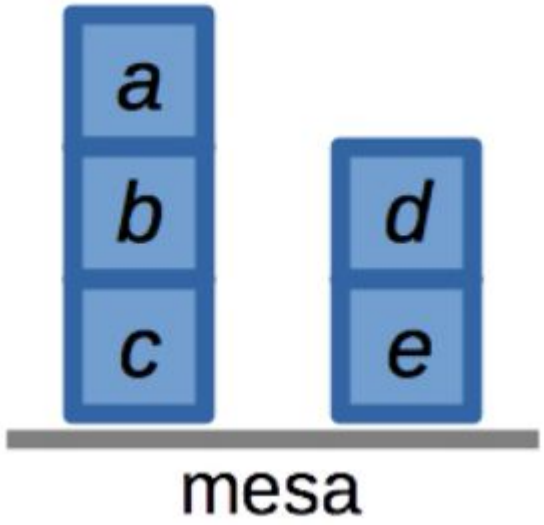
Planificación

- Una de las aplicaciones importantes para la lógica de primer orden es en cuanto a la resolución de problemas con la planificación de sistemas. Para esto es necesario formalizar el problema → "conceptualización":
- Universo de discurso
- Relaciones entre ellos
- Función: describe todos los posibles grupos de objetos entre los que se da una relación específica
- Relacion: determina la falsedad o veracidad de la existencia de esa relación entre un grupo de objetos
- Situación, evento, acción o meta.

El mundo de los cubos

El mundo de los cubos

- Problema clásico de inteligencia artificial Permite
- demostrar el funcionamiento de un sistema de planificación



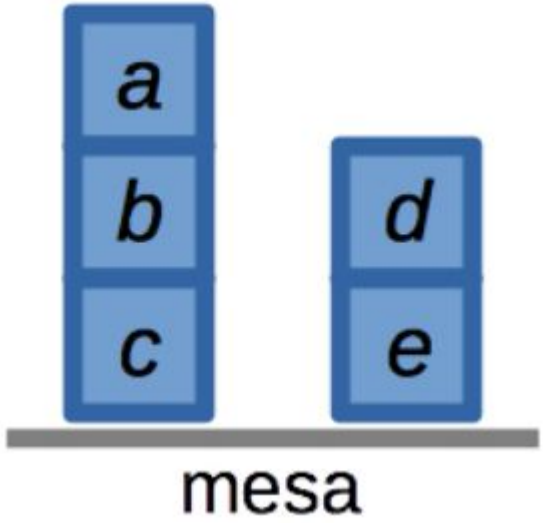
El mundo de los cubos:
Conceptualización

- Universo de discurso

Cubos
Mesa
Brazo robótico

- Relaciones de los objetos (estados)

cubo(x)
mesa(x)
sobre(x, y)
libre(x)
sostiene(x)



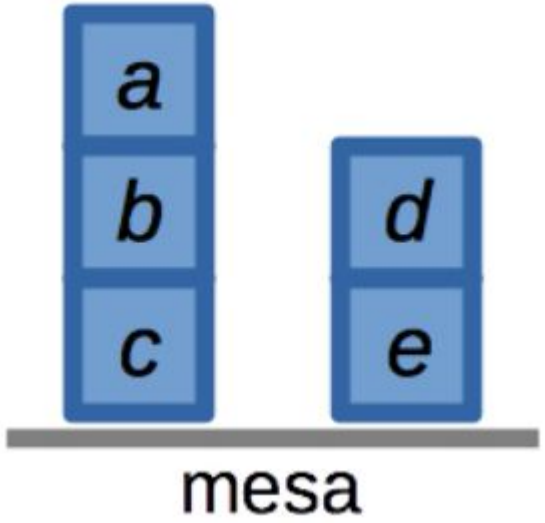
El mundo de los cubos:
Conceptualización

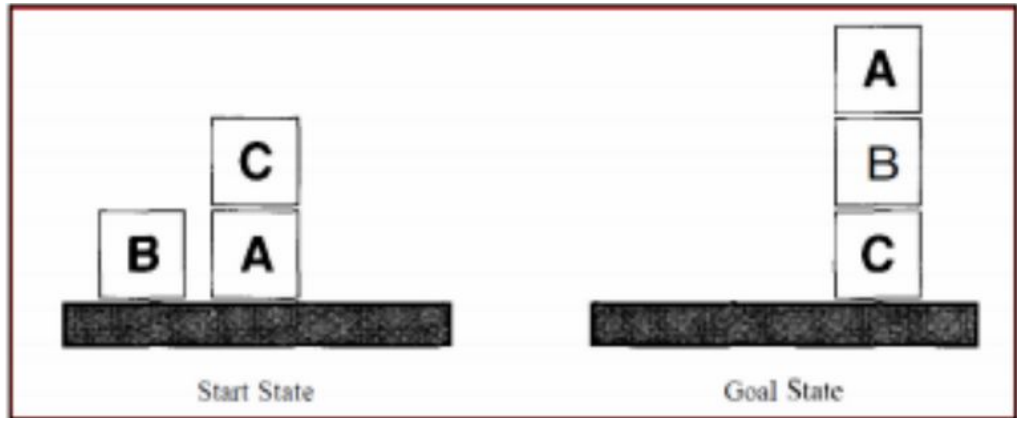
- Operaciones (trancisiones)

agarrarDeLaMesa(x)
agarrarDelCubo(x, y)
ponerEnLaMesa(x)
ponerEnCubo(x, y)

- Meta

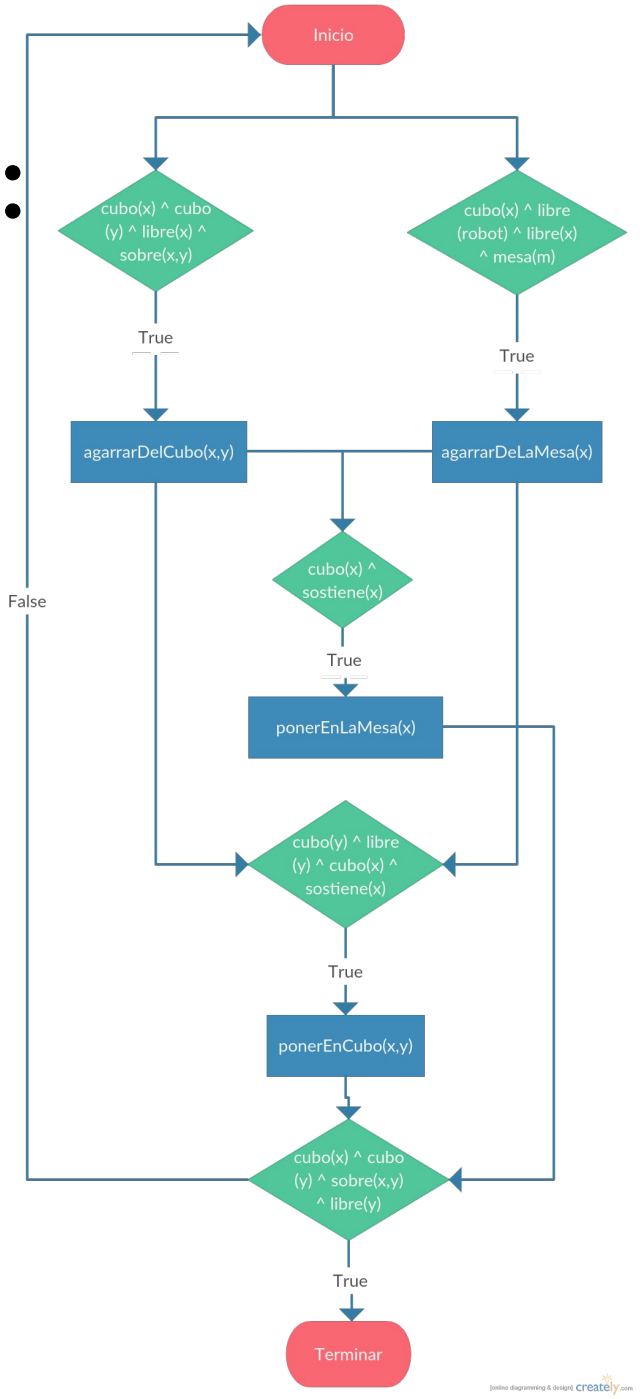
Alguna configuración específica deseada
Como ejemplo tomemos la siguiente





El mundo de los cubos:
Planeación

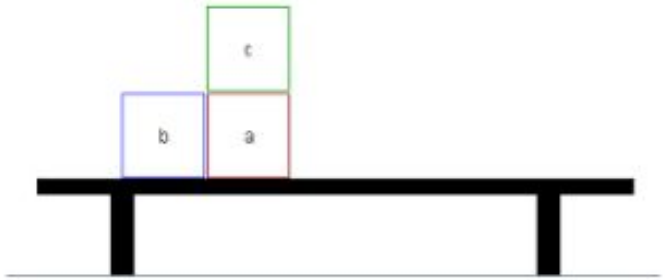
- Una vez se tiene la conceptualización, se puede crear un flujo simple con estos datos que resuelva el problema.
- Este flujo debe tener siempre una condición de inicio, que una vez sea verdadera, se dispara y comienza la ejecución.
- Una vez esto, se resuelve aplicando reglas lógicas simples para:
- Verificar cual es el estado del universo de acuerdo a las relaciones dadas en ese instante.
- Con base a esto, ejecutar la transición adecuada, aplicando los operadores definidos.



El mundo de los cubos:
Ejemplo

Comienzo

Cubo(a) ^ Cubo(b) ^ Cubo(c) ^ Mesa(a) ^ Mesa(b) ^ Libre(b) ^ Libre(c) ^ Sobre (c,a)



Cubo(c) ^ Sobre(c, a) ^ Libre(c) ^ Cubo(a)

AgarrarDelCubo(c)

Sosteniendo(c), Libre(a)

Sosteniendo(c)

PonerEnLaMesa(c)

Libre(c), Mesa(c), Libre(robot)

El mundo de los cubos:
Ejemplo

Estado

Libre(c), Mesa(c), Libre(b), Mesa(b),
Libre(a), Mesa(a), Libre(robot)

Cubo(b) ^
Libre(b) ^ Libre
(robot) ^ Mesa(b)

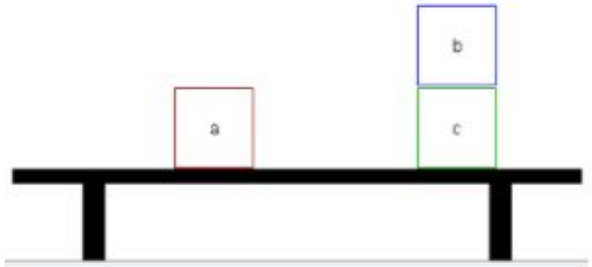
AgarrarDeLaMesa(b)

Sosteniendo(b)

Cubo(c) ^ Libre
(c) ^ Sosteniendo
(b) ^ Cubo(b)

PonerSobreCubo(b,c)

Sobre(b,c) ^ Libre(x) ^ Libre(robot)



El mundo de los cubos: Ejemplo

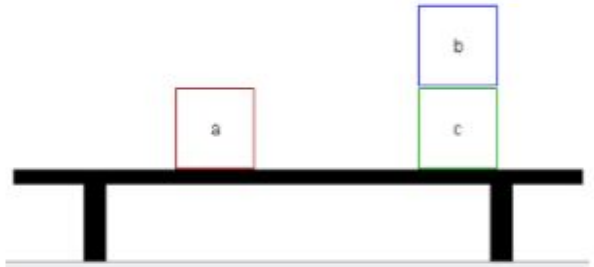
PonerSobreCubo(a,b)

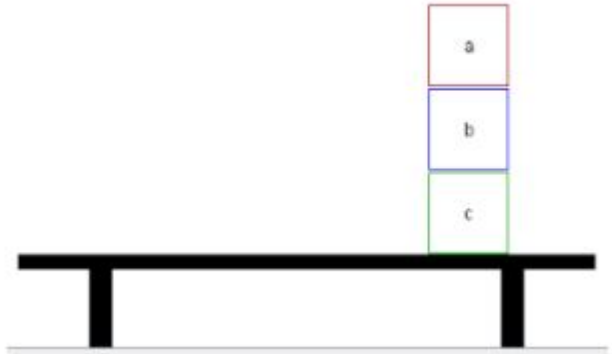
Cubo(a) ^ Cubo(b) ^ Cubo(c) ^ Mesa (c) ^ Sobre(b,c) ^ Sobre(a,b) ^ Libre (a) ^ Libre(robot)

Terminar

Estado

Mesa(c), Libre(b), Mesa(b), Libre(a), Mesa(a), Libre(robot)





Cubo(a) ^ Libre(a) ^ Libre (robot) ^ Mesa(a)

AgarrarDeLaMesa(a)

Sosteniendo(a)

Cubo(b) ^ Libre (b) ^ Sosteniendo (a) ^ Cubo(a)

Referencias

Bernal, L. (2010). “Reglas de inferencia. Lógica de predicados”. Universidad Nacional Abierta y a Distancia. Consultado el 11 de octubre del 2015 en el sitio web: http://es.slideshare.net/leSobreardobernalzamora/reglas-de-inferencia

Fernández, G. (2004). “Representación del conocimiento en sistemas inteligentes”. Capítulo 4 “Lógica de predicados de primer orden”. Universidad Politécnica de Madrid. Consultado el 11 de octubre del 2015 en el sitio web: http://www.gsi.dit.upm.es/~gfer/ssii/rcsi/rcsich4.html

Guerra Hernández, A. (2011). “Lógica de primer orden”. Universidad Veracruzana, Xalapa, México. Consultado el 11 de octubre del 2015 en el sitio web: http://www.uv.mx/aguerra/documents/2011-mpi-02.pdf

Referencias

Guerra Hernández, A. (2011). “Introducción a la programación lógica. En Metodologías Prog. I”. Veracruz, México: Universidad Veracruzana.

Hernández, H. (2008). “Solución de problemas con planificación”. Agentes inteligentes (blog). Consultado el 11 de octubre del 2015 en el sitio web: http://hernandezland.blogspot.com/2008/10/el-mundo-de-los-bloques-estados-del.html

Pavón, N. “Lógica matemática”. Universidad de Huelva. Consultado el 11 de octubre del 2015 en el sitio web: http://www.uhu.es/nieves.pavon/pprogramacion/temario/anexo/anexo.html

Muchas gracias