Cours MALG & MOVEX

MALG

Théorie du Point-Fixe et ses Applications

Dominique Méry
Telecom Nancy, Université de Lorraine
(30 mars 2025 at 3:12 P.M.)

Année universitaire 2024-2025

Plan

Sommaire

Un modèle ensembliste MK pour un système $\mathcal{S}$ est une structure $(\Sigma, \textsc{Init}, \longrightarrow)$ où

- Σ est l'ensemble de tous les états.
- $\mathrm{INIT} \subseteq \Sigma$ définit l'ensemble des états initiaux de S.
- $\longrightarrow$ est une relation binaire

Soit un modèle ensembliste MK pour S.
$\textsc{reachable}(\mathrm{MK}) = \{u | u \in \Sigma \wedge \exists s \in \Sigma. \left(s \in \mathrm{INIT} \wedge s \xrightarrow{\star} u\right)\}$ est l'ensemble de tous les états accessibles pour le modèle ensembliste MK.

Propriété de sûreté

Une propriété A est une propriété de sûreté pour le système S, si

$$\forall i, s \in \Sigma. i \in \mathrm{INIT} \wedge i \xrightarrow{\star} s \Rightarrow s \in A.$$

Invariant pour vérifier une propriété de sûreté

Soit une propriété de sûreté A pour MK :

- $\forall i, s \in \Sigma . i \in \textsc{Init} \wedge i \xrightarrow{\star} s \Rightarrow s \in A$.

  si, et seulement si,

- $\textsc{reachable}(\mathrm{MK}) \subseteq A$

$\textsc{reachable}(\mathrm{MK})$ vérifie les propriétés suivantes :

- $\textsc{Init} \subseteq \textsc{reachable}(\mathrm{MK})$
- $\longrightarrow [\textsc{reachable}(\mathrm{MK})] \subseteq \textsc{reachable}(\mathrm{MK})$ ($\longrightarrow [U]$) est l'ensemble des éléments en relation à U par $\longrightarrow$).
- $\forall U . \begin{pmatrix} U \subseteq \Sigma \\ \textsc{Init} \subseteq U \\ \wedge \\ \longrightarrow [U] \subseteq U \end{pmatrix} \Rightarrow \textsc{reachable}(\mathrm{MK}) \subseteq U$

$\longrightarrow [U] = \{v | v \in \Sigma \wedge \exists u . (u \in U \wedge u \longrightarrow v)\}$

Principe de la vérification exhaustive

Soit une propriété de sûreté A pour MK :

- $\forall i, s \in \Sigma . i \in \textsc{Init} \wedge i \xrightarrow{\star} s \Rightarrow s \in A.$

  si, et seulement si,

- $\textsc{reachable}(\mathrm{MK}) \subseteq A$

- $\textsc{reachable}(\mathrm{MK}) = \textsc{Init} \cup \longrightarrow [\textsc{reachable}(\mathrm{MK})]$

$$\textsc{reachable}(\mathrm{MK}) = \mathrm{FP}(\textsc{reachable}(\mathrm{MK}))$$

où pour toute partie U de Σ, $U = \mathrm{FP}(U)$

- pour toute partie U de Σ, $\mathrm{FP}(U) = Init \cup \longrightarrow [U]$

✓Problématique de la théorie du point fixe

- Résoudre des équations de la forme suivante :

$$X = \mathcal{F}(X)$$

- Donner ou, mieux encore, calculer les solutions, quand elles existent
  - Problème de l'existence de solution selon des hypothèses.
  - Problème de calculer la valeur d'une solution

Théorème du point fixe de Picard

Soit (E, d) un espace métrique complet, et $f : E \longrightarrow E$ une application contractante, c'est-à-dire qu'il existe $k \in [0, 1[$ tel que pour tout $(x, y) \in E$, $d(f(x), f(y)) \leq k \cdot d(x, y)$. Alors f possède un unique point fixe ℓ. De plus, toute suite définie par $u_0 \in I$, $u_{n+1} = f(u_n)$, converge vers cet unique point fixe, et on a les estimations suivantes :

- $d(u_n, \ell) \leq k^n \times d(u_0, l)$
- $d(u_n, \ell) \leq k/(1-k) \times d(u_n, u_{n-1})$

Application I : structure inductive

```
struct liste {
        int val;
        struct liste *next
        }

int longueur(struct liste *l)
        {
        if (l == NIL) {return(0);}
        else {return(1 + longueur(l -> next));}
```

✓Application II : définition récursive

```
#include <stdio.h>
int  f (int x, int y)
{  int i;
   if (x==y)
     {i = y+1;return(i);}
   else
     {i=f(x,f(x-1,y+1));return(i);}
}
int main ()
{
   int a =  2;
   int b = 2;
   printf("Valeur:%d\n",f(a,b));
}
```

✓Définition d'une fonction

```
fun A(0,y) = y + 1
| A(x,0) = A(x-1,1)
| A(x,y) = A(x-1,A(x,y-1));
val A = fn : int * int -> int
```

Propriété à prouver

- $\forall (x,y) \in nat \times nat : (x,y) \in Domaine(A)$

ou

- $\forall (x,y) \in nat \times nat : P(x,y)$ avec
  $P(x,y) \stackrel{def}{=} ((x,y) \in domaine(A))$.

✓Définition d'une fonction

Méthode

1. Choisir une structure bien fondée qui permettra de couvrir nat : $(nat{\times}nat, <_{lex})$
2. Utiliser le principe d'induction complète pour cette structure :
   1. $\forall y \in nat : P(0, y)$
   2. $\forall x \in nat : x \neq 0 : P(x, 0)$
   3. $\forall (x, y) \in nat{\times}nat : x \neq 0 : y \neq 0 : P(x, y)$

✓Définition d'une fonction

- La définition d'une fonction s'identifie à la notion de terminaison ou de correction totale des programmes impératifs.
- La description fonctionnelle d'un programme impératif pose des problèmes de définition d'une fonction pour chaque programme et donc d'une fonctionnelle générale définie pour le langage de programmation de manière structurelle.
- Question : Que signifie l'écriture suivante ?
  $f(x,y) = if\ x = y\ then\ y{+}1\ else\ f(x, f(x{-}1, y{+}1))\ fi$
- Typage, domaines, calculs etc

Une structure partiellement ordonnée $(X, \sqsubseteq)$ est définie par un ensemble X et une relation d'ordre partiel $\sqsubseteq$.

Relation d'ordre partiel $\sqsubseteq$

- $\sqsubseteq \subseteq X \times X$
- $\{(a,a) | a \in X\} \subseteq \sqsubseteq$ (réflexivité)
- $\forall (a,b).((a \in X \wedge b \in X \wedge (a \sqsubseteq b) \wedge (b \sqsubseteq a)) \Rightarrow (a = b))$ (anti-symétrie)
- $(\sqsubseteq ; \sqsubseteq) \subseteq \sqsubseteq$ (transitivité)

Soit une structure partiellement ordonnée $(X, \sqsubseteq)$. Soit une fonction totale F sur X à valeurs dans X. On dit que $fp \in X$ est un point-fixe de F, si $F(fp) = fp$.

Point-fixe fp de F

- $fp \in X$
- $F \in X \longrightarrow X \quad F(fp) = fp$

Structure partiellement ordonnée inductive (ou CPO)

Une structure partiellement ordonnée inductive (ou un CPO Complete Partially Ordered Set) est une structure partiellement ordonnée $(D, \sqsubseteq)$ telle que :

1. D admet un plus petit élément noté $\perp_D$: $\forall d \in D.\perp_D \sqsubseteq d$.
2. tout ensemble dirigé X de D (X est dirigé, si X est non-vide et si $\forall x, y \in X.\exists z \in X.[x \sqsubseteq z \; et \; y \sqsubseteq z]$) admet une borme supérieure dans D.

Chaine (ou structure totalement ordonnée)

Une structure partiellement ordonnée inductive est une chaîne, si :

$$\forall d, d' \in D, d \sqsubseteq d' oud' \sqsubseteq d.$$

Caractérisation d'une structure partiellement ordonnée inductive

Propriété de réduction aux chaines (théorème)

Une structure partiellement ordonnée inductive (ou un CPO Complete Partially Ordered Set, ou une structure partiellement ordonnée complète (inductive)) est une structure partiellement ordonnée $(D, \sqsubseteq)$ telle que :

1. D admet un plus petit élément noté $\perp_D : \forall d \in D. \perp_D \sqsubseteq d$.
2. toute chaine C de D admet une borme supérieure dans D.

- Montrer qu'une structure est une structure partiellement ordonnée inductive revient à ne montrer le propriété que pour les chaines de la structure.
- La démonstration utilise l'axiome du choix qui permet de construire une chaine à partir d'un ensemble dirigé.

✓Exemple 1 de CPO

Flat CPO

Soit D un ensemble et $\perp$ un élément qui n'est pas élément de D. On définit l'ordre partiel sur $D \cup \{\perp\}$ $\sqsubseteq$ comme suit :

$$\begin{array}{l} \forall d \in D \cup \{\perp\}. d \sqsubseteq d \\ \forall d \in D \cup \{\perp\}. \perp \sqsubseteq d \end{array}$$

On notera $D^{\perp} = D \cup \{\perp\}$. $(D^{\perp}, \sqsubseteq)$ est un CPO.

✓Exemple 2 de CPO

Ensemble des fonctions partielles $A \nrightarrow B$

Soit A un ensemble et B un autre ensemble. On notera $\Im(A,B)$ ou $A \nrightarrow B$ l'ensemble des fonctions de A dans B.
$f \sqsubseteq g$ si, et seulement si, $(dom(f) \subseteq dom(g))$ et $(\forall x \in dom(f).f(x) = g(x))$.

Propriété de $(A \nrightarrow B, \sqsubseteq)$

$(A \nrightarrow B, \sqsubseteq)$ est une structure partiellement ordonnée inductive où $\perp_{A \nrightarrow B}$ est la fonction définie nulle part.

✓Exemple 3 de CPO

Ensemble des parties d'un ensemble

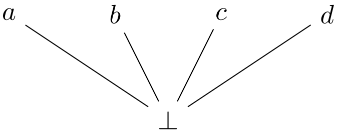
Soit E un ensemble et $\mathcal{P}(E)$, l'ensemble des parties de E. $(\mathcal{P}(E), \subseteq)$ est une structure partiellement ordonnée inductive.

✓Notations

Soit $(A, \sqsubseteq)$ une structure partiellement ordonnée et $D \subseteq A$.

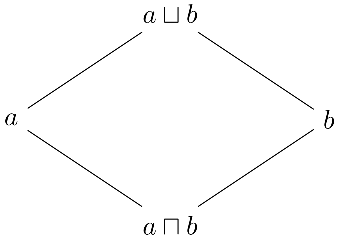
- x est un majorant de D, si $\forall d \in D : d \sqsubseteq x$
- x est un minorant de D, si $\forall d \in D : x \sqsubseteq d$
- La borne supérieure d'une partie de D est le plus petit des majorants.
- La borne inférieure d'une partie de D est le plus grand des minorants.
- $x \sqcup y$ est la borne supérieure de $\{x, y\}$, lorsqu'elle existe.
- $x \sqcap y$ est la borne inférieure de $\{x, y\}$, lorsqu'elle existe.
- $Sup(D) = \sqcup D = \vee D$: borne supérieure de D quand elle existe.
- $Min(D) = \sqcap D = \wedge D$: borne inférieure de D quand elle existe.

✓Illustration des notations



- $D = \{a, b, c, d\}$
- $D^{\perp} = \{a, b, c, d\}$

✓Illustration ds notations



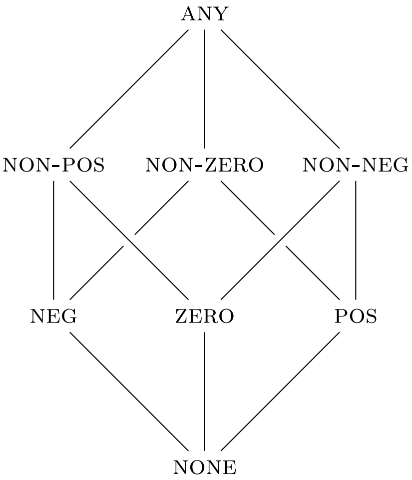
✓Illustration ds notations

▶

✓Illustration ds notations

►

✓Illustration ds notations



- ANY désigne l'ensemble des entiers
- NON-POS désigne l'ensemble des entiers négatifs ou nuls
- POS désigne l'ensemble des entiers positifs non nuls.
- ZERO est l'ensemble contenant uniquement 0.

Croissance d'une fonction

- Une fonction f de $(D_1, \sqsubseteq)$, dans $(D_2, \sqsubseteq)$ est monotone croissante, si :

$$\forall d, d' \in D_1, d \sqsubseteq_1 d' \Rightarrow f(d) \sqsubseteq_2 f(d')$$

- Un ouvert O de D est une partie de D satisfaisant :
  1. Si $x \in O$ et $x \sqsubseteq y$, alors $y \in O$.
  2. Si X est une partie dirigée de D telle que $\sqcup X \in O$, alors $X \cap O \neq \varnothing$.

- Une fonction f est continue pour une topologie donnée, si l'image réciproque de tout ouvert O est un ouvert : si O est un ouvert, $f^{-1}(O)$ est un ouvert.
- Théorème : Soit f une fonction définie de $(D, \sqsubseteq)$ dans $(D', \sqsubseteq')$. f est continue pour la topologie de Scott si, et seulement si, pour tout ensemble dirigé X de D, $f(\sqcup X) = \sqcup f(X)$.
- Une première conséquence du choix de la topologie de Scott est que toute fonction continue est monotone.

Point-fixe pour une fonction continue sur un CPO

Théorème de Kleene

Soit f une fonction continue sur $(D, \sqsubseteq)$ à valeurs dans $(D, \sqsubseteq)$ où $(D, \sqsubseteq)$ est une structure partiellement ordonnée inductive.
Alors il existe un élément x de D tel que

$$\left\{\begin{array}{l} f(x) = x \\ \forall y \in D : (f(y) = y) \Rightarrow (x \sqsubseteq y) \end{array}\right.$$

et on le notera μf.

Preuve du théorème

Preuve du théorème

- $x \stackrel{def}{=} \bigvee_{i \geq 0} f^i$ où $\forall i \in nat^\star : f^{i+1} = f(f^i)$ et $f^0 = \perp_D$:
  x existe car $(D, \sqsubseteq)$ est une structure inductive.

Preuve du théorème

- $x \stackrel{def}{=} \bigvee_{i \geq 0} f^i$ où $\forall i \in nat^\star : f^{i+1} = f(f^i)$ et $f^0 = \perp_D$: x existe car $(D, \sqsubseteq)$ est une structure inductive.
- x est un point-fixe de f :
  $f(x) \stackrel{\textit{definition}}{=} f(\bigvee_{i \geq 0} f^i) \stackrel{\textit{continuite}}{=} \bigvee_{i \geq 0} f(f^i)$
  $\bigvee_{i \geq 0} f(f^i) \stackrel{\textit{definition de la suite}}{=} \bigvee_{i \geq 0} f^{i+1} \stackrel{\textit{renommage}}{=}$
  $\bigvee_{j > 0} f^j \stackrel{\textit{propriete de } \perp_D}{=} \perp_D \sqcup \bigvee_{j > 0} f^j = x$

Preuve du théorème

- $x \stackrel{def}{=} \bigvee_{i \geq 0} f^i$ où $\forall i \in nat^\star : f^{i+1} = f(f^i)$ et $f^0 = \perp_D$:
  x existe car $(D, \sqsubseteq)$ est une structure inductive.
- x est un point-fixe de f :
  $f(x) \stackrel{\textit{definition}}{=} f(\bigvee_{i \geq 0} f^i) \stackrel{\textit{continuite}}{=} \bigvee_{i \geq 0} f(f^i)$
  $\bigvee_{i \geq 0} f(f^i) \stackrel{\textit{definition de la suite}}{=} \bigvee_{i \geq 0} f^{i+1} \stackrel{\textit{renommage}}{=}$
  $\bigvee_{j > 0} f^j \stackrel{\textit{propriete de } \perp_D}{=} \perp_D \sqcup \bigvee_{j > 0} f^j = x$
- x est le plus petit point-fixe de F.
  Soit y un autre point-fixe de f :
  - $\perp_D \sqsubseteq y$
  - Si $\forall j \in \{0, \ldots, i-1\} : f^j \sqsubseteq y$, alors $f^i \sqsubseteq y$.
  - $\forall j \in nat : f^j \sqsubseteq y$

  D'où $x \sqsubseteq y$.
  $\square$

Un exemple de fonction

```
int f5(int x)
{if (x==0)
    { return(0);}
  else
    { if  (x > 0)
 {return(2-f5(1-x));}
      else
 {/* x <0 */
   return(f5(-x));}
    }
}
```

Un exemple de fonction

```
int f5(int x)
{if  (x==0)
    { return(0);}
  else
    { if   (x > 0)
 {return(2-f5(1-x));}
        else
 {/* x <0 */
   return(f5(-x));}
    }
}
```

- Déterminer l'espace du problème : $\mathbb{Z} \nrightarrow \mathbb{Z}$
- Définir la fonctionnelle définissant l'équation du problème : $\mathcal{F} \in (\mathbb{Z} \nrightarrow \mathbb{Z}) \longrightarrow (\mathbb{Z} \nrightarrow \mathbb{Z})$

Un exemple de fonction

```
int f5(int x)
{if (x==0)
    { return(0);}
  else
    { if  (x > 0)
 {return(2-f5(1-x));}
       else
 {/* x <0 */
   return(f5(-x));}
    }
}
```

- Déterminer l'espace du problème : $\mathbb{Z} \nrightarrow \mathbb{Z}$
- Définir la fonctionnelle définissant l'équation du problème : $\mathcal{F} \in (\mathbb{Z} \nrightarrow \mathbb{Z}) \longrightarrow (\mathbb{Z} \nrightarrow \mathbb{Z})$
- $(\mathbb{Z} \nrightarrow \mathbb{Z}, \sqsubseteq)$ est un CPO pour l'ordre *moins défini que*.

Un exemple de fonction

```
int f5(int x)
{if  (x==0)
    { return(0);}
  else
    { if   (x > 0)
 {return(2-f5(1-x));}
        else
 {/* x <0 */
   return(f5(-x));}
    }
}
```

- Déterminer l'espace du problème : $\mathbb{Z} \nrightarrow \mathbb{Z}$
- Définir la fonctionnelle définissant l'équation du problème : $\mathcal{F} \in (\mathbb{Z} \nrightarrow \mathbb{Z}) \longrightarrow (\mathbb{Z} \nrightarrow \mathbb{Z})$
- $(\mathbb{Z} \nrightarrow \mathbb{Z}, \sqsubseteq)$ est un CPO pour l'ordre *moins défini que*.
- $\mathcal{F}$ est continue pour la topologie de Scott

Un exemple de fonction

```
int f5(int x)
{if (x==0)
    { return(0);}
  else
    { if  (x > 0)
 {return(2-f5(1-x));}
        else
 {/* x <0 */
   return(f5(-x));}
    }
}
```

- Déterminer l'espace du problème : $\mathbb{Z} \nrightarrow \mathbb{Z}$
- Définir la fonctionnelle définissant l'équation du problème : $\mathcal{F} \in (\mathbb{Z} \nrightarrow \mathbb{Z}) \longrightarrow (\mathbb{Z} \nrightarrow \mathbb{Z})$
- $(\mathbb{Z} \nrightarrow \mathbb{Z}, \sqsubseteq)$ est un CPO pour l'ordre *moins défini que*.
- $\mathcal{F}$ est continue pour la topologie de Scott
- $f5$ satisfait l'équation $\mathcal{F}(f5) = f5$

Un exemple de fonction

```
int f5(int x)
{if  (x==0)
    { return(0);}
  else
    { if   (x > 0)
 {return(2-f5(1-x));}
        else
 {/* x <0 */
   return(f5(-x));}
    }
}
```

- Déterminer l'espace du problème : $\mathbb{Z} \nrightarrow \mathbb{Z}$
- Définir la fonctionnelle définissant l'équation du problème : $\mathcal{F} \in (\mathbb{Z} \nrightarrow \mathbb{Z}) \longrightarrow (\mathbb{Z} \nrightarrow \mathbb{Z})$
- $(\mathbb{Z} \nrightarrow \mathbb{Z}, \sqsubseteq)$ est un CPO pour l'ordre *moins défini que*.
- $\mathcal{F}$ est continue pour la topologie de Scott
- $f5$ satisfait l'équation $\mathcal{F}(f5) = f5$

$$\boxed{\mathcal{F}(f)(x) = \textbf{if } (x == 0) \textbf{ then } 0 \textbf{ elseif } (x > 0) \textbf{ then } 2-f(1-x)) \textbf{ else } f(-x) \textbf{ fi}}$$

Développer la solution par itération

Développer la solution par itération

- $\mathcal{F}^0 = \{\}$

Développer la solution par itération

- $\mathcal{F}^0 = \{\}$
- $\mathcal{F}^1 = \mathcal{F}(\mathcal{F}^0) = \{(0,0)\}$

Développer la solution par itération

- $\mathcal{F}^0 = \{\}$
- $\mathcal{F}^1 = \mathcal{F}(\mathcal{F}^0) = \{(0,0)\}$
- $\mathcal{F}^2 = \mathcal{F}(\mathcal{F}^1) = \{(0,0),(1,2)\}$

Développer la solution par itération

- $\mathcal{F}^0 = \{\}$
- $\mathcal{F}^1 = \mathcal{F}(\mathcal{F}^0) = \{(0,0)\}$
- $\mathcal{F}^2 = \mathcal{F}(\mathcal{F}^1) = \{(0,0),(1,2)\}$
- $\mathcal{F}^3 = \mathcal{F}(\mathcal{F}^2) = \{(-1,2),(0,0),(1,2)\}$

Développer la solution par itération

- $\mathcal{F}^0 = \{\}$
- $\mathcal{F}^1 = \mathcal{F}(\mathcal{F}^0) = \{(0,0)\}$
- $\mathcal{F}^2 = \mathcal{F}(\mathcal{F}^1) = \{(0,0),(1,2)\}$
- $\mathcal{F}^3 = \mathcal{F}(\mathcal{F}^2) = \{(-1,2),(0,0),(1,2)\}$
- $\mathcal{F}^4 = \mathcal{F}(\mathcal{F}^3) = \{(-1,2),(0,0),(1,2),(2,0)\}$

Développer la solution par itération

- $\mathcal{F}^0 = \{\}$
- $\mathcal{F}^1 = \mathcal{F}(\mathcal{F}^0) = \{(0,0)\}$
- $\mathcal{F}^2 = \mathcal{F}(\mathcal{F}^1) = \{(0,0),(1,2)\}$
- $\mathcal{F}^3 = \mathcal{F}(\mathcal{F}^2) = \{(-1,2),(0,0),(1,2)\}$
- $\mathcal{F}^4 = \mathcal{F}(\mathcal{F}^3) = \{(-1,2),(0,0),(1,2),(2,0)\}$
- $\mathcal{F}^5 = \mathcal{F}(\mathcal{F}^4) = \{(-2,0),(-1,2),(0,0),(1,2),(2,0)\}$

Développer la solution par itération

- $\mathcal{F}^0 = \{\}$
- $\mathcal{F}^1 = \mathcal{F}(\mathcal{F}^0) = \{(0,0)\}$
- $\mathcal{F}^2 = \mathcal{F}(\mathcal{F}^1) = \{(0,0),(1,2)\}$
- $\mathcal{F}^3 = \mathcal{F}(\mathcal{F}^2) = \{(-1,2),(0,0),(1,2)\}$
- $\mathcal{F}^4 = \mathcal{F}(\mathcal{F}^3) = \{(-1,2),(0,0),(1,2),(2,0)\}$
- $\mathcal{F}^5 = \mathcal{F}(\mathcal{F}^4) = \{(-2,0),(-1,2),(0,0),(1,2),(2,0)\}$
- $\mathcal{F}^6 = \mathcal{F}(\mathcal{F}^4) = \{(-2,0),(-1,2),(0,0),(1,2),(2,0),(3,2)\}$

Développer la solution par itération

- $\mathcal{F}^0 = \{\}$
- $\mathcal{F}^1 = \mathcal{F}(\mathcal{F}^0) = \{(0,0)\}$
- $\mathcal{F}^2 = \mathcal{F}(\mathcal{F}^1) = \{(0,0),(1,2)\}$
- $\mathcal{F}^3 = \mathcal{F}(\mathcal{F}^2) = \{(-1,2),(0,0),(1,2)\}$
- $\mathcal{F}^4 = \mathcal{F}(\mathcal{F}^3) = \{(-1,2),(0,0),(1,2),(2,0)\}$
- $\mathcal{F}^5 = \mathcal{F}(\mathcal{F}^4) = \{(-2,0),(-1,2),(0,0),(1,2),(2,0)\}$
- $\mathcal{F}^6 = \mathcal{F}(\mathcal{F}^4) = \{(-2,0),(-1,2),(0,0),(1,2),(2,0),(3,2)\}$
- $\mathcal{F}^7 = \mathcal{F}(\mathcal{F}^4) = \{(-3,0),(-2,0),(-1,2),(0,0),(1,2),(2,0),(3,2)\}$

Développer la solution par itération

- $\mathcal{F}^0 = \{\}$
- $\mathcal{F}^1 = \mathcal{F}(\mathcal{F}^0) = \{(0,0)\}$
- $\mathcal{F}^2 = \mathcal{F}(\mathcal{F}^1) = \{(0,0),(1,2)\}$
- $\mathcal{F}^3 = \mathcal{F}(\mathcal{F}^2) = \{(-1,2),(0,0),(1,2)\}$
- $\mathcal{F}^4 = \mathcal{F}(\mathcal{F}^3) = \{(-1,2),(0,0),(1,2),(2,0)\}$
- $\mathcal{F}^5 = \mathcal{F}(\mathcal{F}^4) = \{(-2,0),(-1,2),(0,0),(1,2),(2,0)\}$
- $\mathcal{F}^6 = \mathcal{F}(\mathcal{F}^4) = \{(-2,0),(-1,2),(0,0),(1,2),(2,0),(3,2)\}$
- $\mathcal{F}^7 = \mathcal{F}(\mathcal{F}^4) = \{(-3,0),(-2,0),(-1,2),(0,0),(1,2),(2,0),(3,2)\}$
- $\ldots$

Développer la solution par itération

- $\mathcal{F}^0 = \{\}$
- $\mathcal{F}^1 = \mathcal{F}(\mathcal{F}^0) = \{(0,0)\}$
- $\mathcal{F}^2 = \mathcal{F}(\mathcal{F}^1) = \{(0,0),(1,2)\}$
- $\mathcal{F}^3 = \mathcal{F}(\mathcal{F}^2) = \{(-1,2),(0,0),(1,2)\}$
- $\mathcal{F}^4 = \mathcal{F}(\mathcal{F}^3) = \{(-1,2),(0,0),(1,2),(2,0)\}$
- $\mathcal{F}^5 = \mathcal{F}(\mathcal{F}^4) = \{(-2,0),(-1,2),(0,0),(1,2),(2,0)\}$
- $\mathcal{F}^6 = \mathcal{F}(\mathcal{F}^4) = \{(-2,0),(-1,2),(0,0),(1,2),(2,0),(3,2)\}$
- $\mathcal{F}^7 = \mathcal{F}(\mathcal{F}^4) = \{(-3,0),(-2,0),(-1,2),(0,0),(1,2),(2,0),(3,2)\}$
- $\ldots$
- $g(x) = \textbf{if } odd(x) \textbf{ then } 2 \textbf{ else } 0 \textbf{ fi}$ est solution de cette équation et $\mu\mathcal{F} \sqsubseteq g$ mais on doit montrer que $\mu\mathcal{F} = g$

- Un prédicat P sur un ensemble inductif $(D, \sqsubseteq)$ (CPO) est admissible, si, pour tout ensemble dirigé A de D, si, pour tout a de A $P(a)$, alors $P(\bigvee A)$.
- Principe d'induction du point fixe :
  Soit $(D, \sqsubseteq)$ une structure inductive, P un prédicat admissible sur D, F une fonction continue de D dans D.
  Si
  1. $P(\perp_D)$
  2. Soit $d \in D$ tel que $P(d)$:

$\vdots$

$P(F(d))$.

Alors $P(\mu F)$.

Prédicats admissibles

- $P(x)$ défini par $x = a$ est admissible.
- $P(x)$ défini par $x \sqsubseteq a$ est admissible.
- Si f_i et g_i sont deux suites de fonctions continues sur un CPO $(D, \sqsubseteq)$, alors $\bigwedge_i f_i(x) \sqsubseteq g_i(x)$ est un prédicat admissible.

Fonction 91 de MacCarthy

- $F(f)(x) = si\ 100 < x\ alors\ x{-}10\ sinon\ f(f(x{+}11))\ fi$ et $g(x) = si\ 100 < x\ then\ x{-}10\ sinon\ 91\ fi$. Montrez que $\mu F \sqsubseteq g$.

Fonction 91 de MacCarthy

- $F(f)(x) = si\ x > 100\ alors\ x{-}10\ sinon\ f(f(x{+}11))\ fi$
- et $g(x) = si\ x > 100\ then\ x{-}10\ sinon\ 91\ fi$.
- On peut montrer que $\mu f \sqsubseteq g$.
- Puis on peut en déduire que $\mu f = g$, en montrant que la fonction μf est totale.

Fonction f5

$$\boxed{\mathcal{F}(f)(x) = \textbf{if } (x == 0) \textbf{ then } 0 \textbf{ elseif } (x > 0) \textbf{ then } 2 - f(1-x)) \textbf{ else } f(-x) \textbf{ fi}}$$

- $g(x) = \textbf{if } odd(x) \textbf{ then } 2 \textbf{ else } 0 \textbf{ fi}$ est solution de cette équation
- Utilisation de $P(f) \stackrel{def}{=} f \sqsubseteq g$
- Montrer que $dom(\mu f) = \mathbb{N}$:
  - $\mathcal{F}^{2n} = \{(p, v_p) | 0 \leq p < n \wedge v_p = g(p)\} \cup \{(p, v_p) | 0 > p \geq -(n-1) \wedge v_p = g(p)\} \cup \{(n, g(n))\}$
  - $\mathcal{F}^{2n+1} = \{(p, v_p) | 0 \leq p < n \wedge v_p = g(p)\} \cup \{(p, v_p) | 0 > p \geq -(n-1) \wedge v_p = g(p)\} \cup \{(n, g(n)), (-n, g(-n))\}$
  - $\mu\mathcal{F} = \mathcal{F}^0 \cup \mathcal{F}^1 \cup \mathcal{F}^2 \cup \mathcal{F}^3 \cup \ldots \cup \mathcal{F}^{2n} \cup \mathcal{F}^{2n+1} \ldots$
  - Pour tout $p \in \mathbb{N}$, $p \in \mathcal{F}^{2p} \cup \mathcal{F}^{2p+1}$ par construction de cette suite.

Treillis complet

⊠ Definition

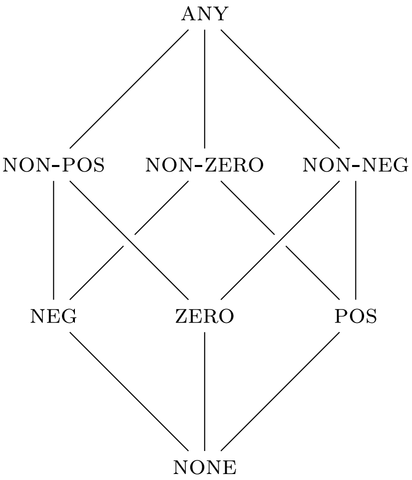
Un treillis complet $(L, \sqsubseteq)$ est un treillis satisfaisant les propriétés suivantes :

1. Pour toute partie A de L, il existe une borne supérieure notée $\sqcup A$.
2. Pour toute partie A de L, il existe une borne inférieure notée $\sqcap A$.

☼ Exemple

1. Un exemple simple est la structure $(\mathcal{P}(E), \subseteq, \varnothing, \cup, E, \cap)$ associée à l'ensemble des parties d'un ensemble E.
2. Treillis des signes est un treillis complet
$\mathcal{S}igns =$
$\{\text{NONE}, \text{ZERO}, \text{NON-ZERO}, \text{ANY}, \text{POS}, \text{NEG}, \text{NON-NEG}, \text{NON-POS}\}$
$(\mathcal{S}igns, \sqsubseteq)$
3. Treillis des intervalles
   - $\mathbb{I}(\mathbb{Z}) = \{\bot\} \cup \{[l, u] | l \in \mathbb{Z} \cup \{-\infty\}, u \in \mathbb{Z} \cup \{\infty\}, l \leq u\}$
   - $[l_1, u_1] \sqsubseteq [l_2, u_2]$ si, et seulement si, $l2 \leq l1$ et $u_1 \leq u_2$.
   - $(\mathbb{I}(\mathbb{Z}), \sqsubseteq)$ est un treillis complet

Treillis des signes



- ANY désigne l'ensemble des entiers
- NON-POS désigne l'ensemble des entiers négatifs ou nuls
- POS désigne l'ensemble des entiers positifs non nuls.
- ZERO est l'ensemble contenant uniquement 0.

Treillis des intervalles

- $\mathbb{I}(\mathbb{Z}) = \{\perp\} \cup \{[l,u] | l \in \mathbb{Z} \cup \{-\infty\}, u \in \mathbb{Z} \cup \{\infty\}, l \leq u\}$
- $[l_1, u_1] \sqsubseteq [l_2, u_2]$ si, et seulement si, $l2 \leq l1$ et $u_1 \leq u_2$.
- $(\mathbb{I}(\mathbb{Z}), \sqsubseteq)$ est une structure partiellement ordonnée.
-
  1. $[l_1, u_1] \sqcup [l_2, u_2] = [min(l_1, l_2), max(u_1, u_2)]$
  2. $[l_1, u_1] \sqcap [l_2, u_2] = \begin{cases} [max(l_1, l_2), min(u_1, u_2)] \\ \perp, si\ max(l_1, l_2) > min(u_1, u_2) \end{cases}$
- $(\mathbb{I}(\mathbb{Z}), \sqcup)$ est un treillis complet.

Treillis complet

. . . .

⊠ Definition

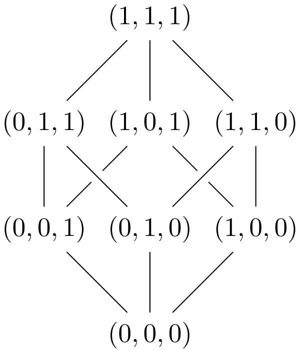
Un treillis complet $(L, \sqsubseteq, \bot, \sqcup, \top, \sqcap)$ est un treillis satisfaisant les propriétés suivantes :

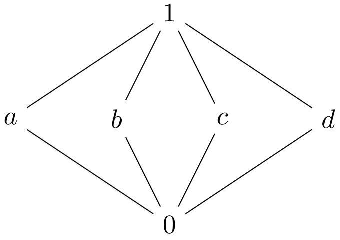
1. Pour toute partie A de L, il existe une borne supérieure notée $\sqcup A$.
2. Pour toute partie A de L, il existe une borne inférieure notée $\sqcap A$.

. . . .

- Un treillis complet $(L, \sqsubseteq,)$ peut être défini par les éléments suivants $(L, \sqsubseteq, \bot, \sqcup, \top, \sqcap)$
- Un treillis est une structure munie d'un ordre partiel et telle que deux éléments ont une borne supérieure et une inférieure dans le treillis : $(L, \sqsubseteq, \sqcup, \sqcap)$
  - $a \sqcup b$ existe et est la borne supérieure des deux éléments a et b.
  - $a \sqcap b$ existe et est la borne inférieure des deux éléments a et b.

Représentation par des diagrammes de Hasse





Pre/Post Points-fixes

Pour une fonction f définie sur un treillis $(L, \sqsubseteq, \bot, \sqcup, \top, \sqcap)$ et à valeurs dans $(L, \sqsubseteq, \bot, \sqcup, \top, \sqcap)$.

Définition

- on appelle pré-point-fixe de f, tout élément x de L satisfaisant la propriété $f(x) \sqsubseteq x$
- on appelle post-point-fixe de f, tout élément x de L satisfaisant $x \sqsubseteq f(x)$.

- $PostFIX(f) = \{x | x \in L \wedge x \sqsubseteq f(x)\}$: l'ensemble des post-points-fixes de f.
- $PreFIX(f) = \{x | x \in L \wedge f(x) \sqsubseteq x\}$: l'ensemble des pré-points-fixes de f.
- $FIX(f) = \{x | x \in L \wedge f(x) = x\}$: l'ensemble des points-fixes de f.

Théorème de Knaster-Tarski (I)

Soit f une fonction monotone croissante sur un treillis complet $(T, \bot, \top, \bigvee, \bigwedge)$. Alors il existe un plus petit point fixe et un plus grand point fixe pour f.

- μf désigne le plus petit point fixe de f.
- νf désigne le plus grand point fixe de f.
- μf vérifie les propriétés suivantes : $f(\mu f) = \mu f$ et $\forall x \in T. f(x) \sqsubseteq x \Rightarrow \mu f \sqsubseteq x$ (μf est la borne inférieure des prépoints fixes de f ou $\bigwedge(Pre(f))$).
- νf vérifie les propriétés suivantes : $f(\nu f) = \nu f$ et $\forall x \in T. x \sqsubseteq f(x) \Rightarrow x \sqsubseteq \nu f$ (νf est la borne supérieure des postpoints fixes de f ou $\bigvee(Post(f))$).

Preuve

Posons $y = \bigwedge\{x|f(x) \sqsubseteq x\}$ et montrons que y est un point fixe de f et que y est le plus petit point fixe de f.

1. $f(y) \sqsubseteq y$
   - Pour tout x de $\{x|f(x) \sqsubseteq x\}$, $y \sqsubseteq x$
   - $f(y) \sqsubseteq f(x)$ (par monotonie de f).
   - $f(x) \sqsubseteq x$ (par définition de x).
   - $f(y) \sqsubseteq x$ (par déduction).
   - $f(y) \sqsubseteq \bigwedge\{x|f(x) \sqsubseteq x\}$ (par définition de la borne inférieure, $f(y)$ est un minorant).
   - $f(y) \sqsubseteq y$
2. $y \sqsubseteq f(y)$
   - $f(y) \sqsubseteq y$ (par le cas 1)
   - $f(f(y)) \sqsubseteq f(y)$ (par monotonie de f)
   - $f(y) \in \{x|f(x) \sqsubseteq x\}$
   - $y \sqsubseteq f(y)$ (par définition de la borne inférieure)
3. Conclusion : $f(y) = y$ ou $y \in FIX(f)$.

Preuve

- $f(y) = y$ et z tel que $f(z) = z$
  - $f(z) = z$ (par hypothèse sur z)
  - $f(z) \sqsubseteq z$ (par affaiblissement de l'égalité)
  - $z \in \{x | f(x) \sqsubseteq x\}$ (par définition de cet ensemble)
  - $y \sqsubseteq z$ (par construction)
- y est le plus petit point fixe de f.

lfp(f) et gfp(f)

- $\mu f = lfp(f) = \bigwedge\{x | f(x) \sqsubseteq x\}$
- $\nu f = gfp(f) = \bigvee\{x | x \sqsubseteq f(x)\}$

- $lfp(f)$ signifie *least fixed-point*
- $gfp(f)$ signifie *greatest fixed-point*

Positionnement des éléments

$$\begin{array}{c}
\top \\
| \\
x \in \{x|f(x) \sqsubseteq x\} \\
| \\
\nu f \\
| \\
x \in \{x|x = f(x)\} \\
| \\
\mu f \\
| \\
x \in \{x|x \sqsubseteq f(x)\} \\
| \\
\bot
\end{array}$$

- $\top = \sqcup\{x|f(x) \sqsubseteq x\}$
- $gfp(f) = \nu f = \sqcup\{x|x \sqsubseteq f(x)\}$
- $lfp(f) = \mu f = \sqcap\{x|f(x) \sqsubseteq x\}$
- $\bot = \sqcap\{x|x \sqsubseteq f(x)\}$

Preuve algébrique selon Hoare

- $x \in pre(f)$ **si, et seulement si,** $f(x) \leq x$ (définition)

Preuve algébrique selon Hoare

- $x \in pre(f)$ **si, et seulement si,** $f(x) \leq x$ (définition)
- $\forall a \in pre(f).\mu f \leq a$ (application de la définition de la borne inférieure d'un ensemble)

Preuve algébrique selon Hoare

- $x \in pre(f)$ **si, et seulement si,** $f(x) \leq x$ (définition)
- $\forall a \in pre(f).\mu f \leq a$ (application de la définition de la borne inférieure d'un ensemble)
- $\forall a \in pre(f).f(\mu f) \leq f(a) \leq a$ (croissance de f et propriété de a)

Preuve algébrique selon Hoare

- $x \in pre(f)$ **si, et seulement si,** $f(x) \leq x$ (définition)
- $\forall a \in pre(f).\mu f \leq a$ (application de la définition de la borne inférieure d'un ensemble)
- $\forall a \in pre(f).f(\mu f) \leq f(a) \leq a$ (croissance de f et propriété de a)
- $f(\mu f) \leq \mu f$ (application pour $a = \mu f$))

Preuve algébrique selon Hoare

- $x \in pre(f)$ **si, et seulement si,** $f(x) \leq x$ (définition)
- $\forall a \in pre(f).\mu f \leq a$ (application de la définition de la borne inférieure d'un ensemble)
- $\forall a \in pre(f).f(\mu f) \leq f(a) \leq a$ (croissance de f et propriété de a)
- $f(\mu f) \leq \mu f$ (application pour $a = \mu f$))
- $f(f(\mu f)) \leq f(\mu f)$ (croissance de f)

Preuve algébrique selon Hoare

- $x \in pre(f)$ **si, et seulement si,** $f(x) \leq x$ (définition)
- $\forall a \in pre(f).\mu f \leq a$ (application de la définition de la borne inférieure d'un ensemble)
- $\forall a \in pre(f).f(\mu f) \leq f(a) \leq a$ (croissance de f et propriété de a)
- $f(\mu f) \leq \mu f$ (application pour $a = \mu f$))
- $f(f(\mu f)) \leq f(\mu f)$ (croissance de f)
- $f(\mu f) \in pre(f)$ (définition des pré-points-fixes)

Preuve algébrique selon Hoare

- $x \in pre(f)$ **si, et seulement si,** $f(x) \leq x$ (définition)
- $\forall a \in pre(f).\mu f \leq a$ (application de la définition de la borne inférieure d'un ensemble)
- $\forall a \in pre(f).f(\mu f) \leq f(a) \leq a$ (croissance de f et propriété de a)
- $f(\mu f) \leq \mu f$ (application pour $a = \mu f$))
- $f(f(\mu f)) \leq f(\mu f)$ (croissance de f)
- $f(\mu f) \in pre(f)$ (définition des pré-points-fixes)
- $\mu f \leq f(\mu f)$ (définition de μf)
- $\mu f = f(\mu f)$ (définition de μf et propriété précédente)

Version constructive du théorème de Knaster-Tarski

Soit f une fonction monotone croissante sur un treillis complet $(T, \perp, \top, \bigvee, \bigwedge)$. Alors

1. La structure formée des points fixes de f sur T, $(fp(f), \sqsubseteq)$ est un treillis complet non-vide.
   $(fp(f) = \{x \in T : f(x) = x\}$
2. $lfp(f) \stackrel{def}{=} \bigvee_{\alpha} f^{\alpha}$ est le plus petit point fixe de f où :
   $$\left\{ \begin{array}{lccc} & f^0 & \stackrel{def}{=} & \perp \\ \alpha \textit{ ordinal successeur} & f^{\alpha+1} & \stackrel{def}{=} & f(f^{\alpha}) \\ \alpha \textit{ ordinal limite} & f^{\alpha} & \stackrel{def}{=} & \bigvee_{\beta<\alpha} f^{\beta} \end{array} \right.$$
3. $gfp(f) \stackrel{def}{=} \bigwedge_{\alpha} f_{\alpha}$ est le plus grand point fixe de f où
   $$\left\{ \begin{array}{lccc} & f_0 & \stackrel{def}{=} & \top \\ \alpha \textit{ ordinal successeur} & f_{\alpha+1} & \stackrel{def}{=} & f(f_{\alpha}) \\ \alpha \textit{ ordinal limite} & f_{\alpha} & \stackrel{def}{=} & \bigwedge_{\beta<\alpha} f_{\beta} \end{array} \right.$$

- $\mathcal{F}$ est la classe de toutes les fonctions
- La théorie des fonctions récursives concerne l'étude des fonctions dites **calculables**.
- Une méthode de calcul $\mathcal{M}$ est définie par une syntaxe permettant de représenter les données et un ensemble de règles de transformations sur les données : le modèle des machines de Turing $\mathcal{T}$, le modèle des grammaires $\mathcal{G}$, le modèle des Unlimited Register Machines URM $\mathcal{U}$, le modèle d'un langage de programmation $\mathcal{P}$, ...
- Une partielle fonction $f \in A \nrightarrow B$ est calculable, s'il existe une procédure effective ou automatisée P selon la méthode $\mathcal{M}$ telle que

$$\forall a \in dom(f) : \forall b \in B : f(a) = b \;\Leftrightarrow\; a \xrightarrow{\mathsf{P}} b \tag{1}$$

- La classe des fonctions calculables est notée $\mathcal{C}$.

Thèse de Church-Turing (point de vue épistémologique)

Source bibliographique

Les deux formes de la thèse de Church-Turing et l'épistémologie du calcul par Maël Pégny

Deux formes de la thèse

- forme algorithmique : toute fonction calculable par un algorithme est calculable par une machine de Turing :

Theorem

(équivalence des modèles de calcul) $\mathcal{C} = \mathcal{R} = \mathcal{T} = \mathcal{R} = \mathcal{G}$

- forme empirique : toute fonction calculable par une machine est calculable par une machine de Turing.

Question

L'ensemble des fonctions calculables par une machine est-il identique à l'ensemble des fonctions calculables par un algorithme ?
S'il existait une fonction calculable par une machine sans être calculable par un algorithme, il existerait un problème mathématique qui serait soluble par un dispositif empirique, sans être soluble par aucune méthode mathématique a priori.

Théorie des fonctions récursives

- $\mathcal{F}$ est la classe des fonctions partielles sur l'ensemble des naturels $\mathbb{N}$: si $f \in \mathcal{F}$, alors il existe n et m tels que $dom(f) \subseteq \mathbb{N}^n$ et $ran(f) \subseteq \mathbb{N}^m$.
- ($\mathcal{F}$,$\sqsubseteq$)) est une structure partiellement ordonnée inductive où $\sqsubseteq$ est la relation *moins définie que*

$$f \sqsubseteq g \textit{ si, et seulement si, } \begin{cases} dom(f) \subseteq dom(g) \\ \forall x \in dom(f) : f(x) = g(x) \end{cases}$$

Opérateur récursif sur $\mathcal{F}$,

Soient $\mathcal{F}_i = \mathbb{N}^i \nrightarrow \mathbb{N}$ et $f \in \mathcal{F}_m \nrightarrow \mathcal{F}_n$ une fonction partielle. On dit que f est récursif, s'il existe une fonction calculable $h(z, x$ telle que
pour toute fonction $f \in \mathcal{F}_m$, pour tout x de . $\mathbb{N}^n$, $y \in \mathbb{N}$,
$h(z, x) = y$ *si, et seulement si,* il existe une finie onction à domaine fini $g \subseteq f$ satisfaisant $h(\dot{g}, x) = y$

$\dot{g}$ est la valeur associée à g et définie par

$$\dot{g} = \begin{cases} \prod_{i \in dom(g)} {p_{<x>}}^{g(x)+1} \textbf{ si } dom(g) \neq \varnothing \\ 0 \textbf{ si } dom(g) = \varnothing \end{cases}$$

$$< x >= p_1^{x_1+1} + p_2^{x_2+1} + \ldots + p_n^{x_n+1}$$

☺ Théorème

Soient $\mathcal{F}_i = \mathbb{N}^i \nrightarrow \mathbb{N}$ et $f \in \mathcal{F}_m \nrightarrow \mathcal{F}_n$ une fonction partielle.
f est récursif *si, et seulement si,*

1. f est continue pour l'ordre $\sqsubseteq$.
2. la fonction h définie par : $\begin{cases} h(\dot{g}, x) = f(g)(x) \textbf{ si } g \ \in \mathcal{F}_m \\ h(z, x) \ \textit{indefinie} \end{cases}$ est calculable.

☺ Théorème

Soit $f \in \mathcal{F}_m \nrightarrow \mathcal{F}_m$ un opérateur récursif. Alors il existe un élément x de $\mathcal{F}_m$ tel que

$\begin{cases} f(x) = x \\ \forall y \in \mathcal{F}_m : (f(y) = y) \Rightarrow (x \sqsubseteq y) \end{cases}$ et x est calculable.

Lemme de Arden

- Soit l'ensemble des parties de $\Sigma^\star$ munies des opérations classiques des ensembles et l'opération de concaténation.
- La fonction $\mathcal{F}$ définie sur cet ensemble à valeur dans le même ensemble associe à tout ensemble un ensemble obtenu par application des opérations ensemblistes et de la concaténation : $\mathcal{F}(X) = A.X \cup B$
- $(\mathcal{P}(\Sigma^\star), \varnothing, \Sigma^\star, \cup, \cap)$ est un treillis complet.
- $\mathcal{F}$ admet un plus petit point fixe qui est le langage régulier caractérisé par l'expression régulière $a^\star b$

Fonctions induites par les grammaires

- Soit une grammaire algébrique $G = (N, T, P, S)$ définissant un langage sur T.
- On définit pour cette grammaire un opérateur sur $T^\star$ l'ensemble desmots finis sur T, noté $\mathcal{F}_G$ comme suit :
  - $N = \{X_1, \ldots, X_n\}$
  - $S = X_1$ et $X = (X_1, \ldots, X_n)$.
  - $\mathcal{F}_G(X)$ est construit par application des règles de P pour chaque non terminal X_i
  - Si $X_i \longrightarrow v_1 + \ldots + v_p$, on obtient l'ensemble par remplaceme,nt des X_k dans les v_p.
- Exemple :
  - $N = \{X, Y, Z\}$
  - $T = \{a, b\}$
  - $P = \{X \longrightarrow aYZb, Y \longrightarrow aY, Z \longrightarrow Zb, Y \longrightarrow a, Z \longrightarrow b\}$
  - $F(X, Y, Z) = \begin{pmatrix} aYZb \\ aY \cup \{a\} \\ Zb \cup \{b\} \end{pmatrix}$

- F est monotone croisssante sur $(\{X, Y, Z, a, b\}^\star)^3$.
- $\left(\begin{array}{l} X = aYZb \\ Y = aY \cup \{a\} \\ Z = Zb \cup \{b\} \end{array}\right)$
- μF est le plus petit point fixe.
- $L(G) = \pi_1^3(\mu F)$

Définition inductive

- L'ensemble $\mathcal{EVEN}$ des nombres naturels pairs est le plus petit ensemble stable par application des règles suivantes :
  1. $0 \in \mathcal{EVEN}$
  2. Si $n \in \mathcal{EVEN}$, alors $n{+}2 \in \mathcal{EVEN}$
- Opérateur induit : $\mathcal{F}(X) = \{0\} \cup \{n{+}2 | n \in X\}$
- $\mathcal{F} \in \mathbb{P}(\mathbb{N}) \longrightarrow \mathbb{P}(\mathbb{N})$
- $\mathcal{F}(\mathcal{EVEN}) = \mathcal{EVEN}$ et $\mathcal{EVEN} = \mu\mathcal{F}$.

Définition inductive

- Une définition inductive $\mathcal{I}$ est une structure $(U, \mathcal{F}, \bot, \sqcup)$ où
  - U est un ensemble appelé l'univers
  - $\mathcal{R}$ est un ensemble de règles de la forme $\mathbf{Si}\ P, \mathbf{alors}\ C$ où $P \subseteq U$ et $C \in U$.
  - $\bot \in U$
  - $\sqcup \in \mathbb{P}(\mathbb{P}(U)) \longrightarrow \mathbb{P}(U)$
- Pour une définition inductive, on peut dériver un opérateur induit $\mathcal{F}$: $\mathcal{F}(X) = \{c \in U | \exists P \subseteq X : \mathbf{Si}\ P, \mathbf{alors}\ C \in \mathcal{R}\}$
- $\mathcal{F}_1(X) = \{0\} \cup \{n+3 \in \mathbb{N} | n \in X\}$
- $\mathcal{F}_2(X) = Init \cup \{u \in U | \exists s.s \xrightarrow{\star} u \wedge s \in X\}$

Principe d'induction sur les naturels

$\forall A.A \subseteq \mathbb{N} \wedge 0 \in A \wedge suc[A] \subseteq A \Rightarrow \mathbb{N} \subseteq A)$

- $0 \in A$
- Pas d'induction
  - Hypothèse $n \in A$
  - Conclusion $suc(n) \in A$
- Conclusion $\forall n.n \in \mathbb{N} \Rightarrow n \in A$

Soit la propriété suivante à démontrer $\forall n.n \in \mathbb{N} \Rightarrow n \in P(n)$:

1. $A \stackrel{def}{=} \{n | n \in \mathbb{N} \wedge P(n)\}$
2.
   - $0 \in A$
   - Pas d'induction
     - Hypothèse $n \in A$
     - Conclusion $suc(n) \in A$
   - Conclusion $\forall n.n \in \mathbb{N} \Rightarrow n \in A$

Questions et commentaires

- Théorèmes de point-fixe :
  - Théorème de Kleene pour les fonctions récursives ou calculables
  - Définition d'un langage par un système d'équations
  - Calcul de ces points-fixes
- Applications de ces résultats :
  - Analyse des structures inductives
  - Sémantique des langages de programmation
  - Fondements de l'analyse sémantique des langages de programmation et techniques d'analyse automatisée