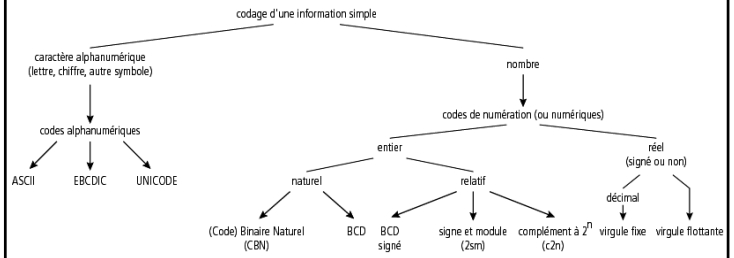
Architecture des ordinateurs

7 - Représentation des réels en machine

Philippe Darche
IUT Paris Descartes

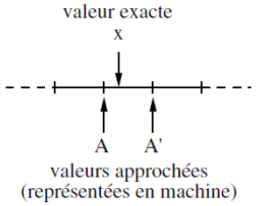
Représentation de l'information en machine (rappel)

- Des 0 et des 1 !
  - B = 2 (base binaire)



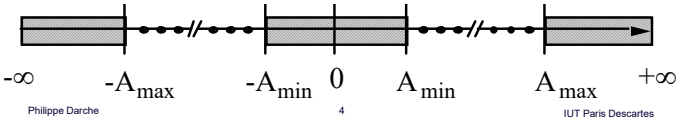
Représentations d'un nombre réel

- La problématique majeur
  - risque d'une représentation approchée
    - ce problème n'existe pas pour un entier !



Représentations d'un nombre réel

- A cela s'ajoute le problème classique d'une représentation qui est le risque d'un (sur-) dépassement de capacité (*overflow*) qui peut être positif ou négatif
- Mais il y a aussi le risque d'un sous-dépassement, soupassement ou dépassement inférieur ou par le bas de capacité (*underflow*)) qui peut être positif ou négatif

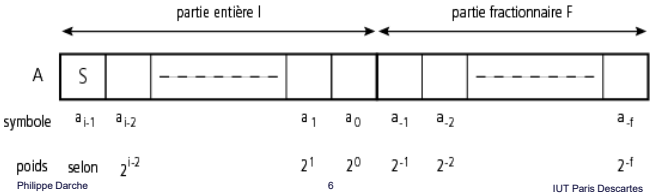


Représentations des réels en machine

- Deux familles
  - représentation en virgule fixe (2vfx)
    - *fixed-point representation* en anglais
    - pour la représentation des décimaux
    - domaine d'utilisation : le traitement numérique du signal principalement (son, vidéo, radio-fréquence)
  - représentation en virgule flottante (2vfl)
    - *floating-point representation* en anglais
    - pour la représentation des réels
    - domaine d'utilisation : le calcul scientifique

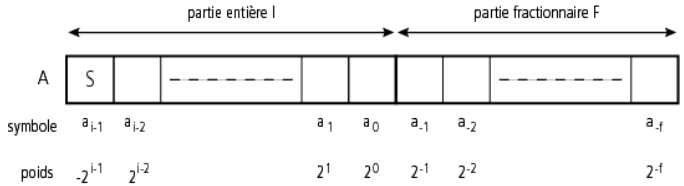
Représentation en virgule fixe

- Une partie entière I au format i, signée dans la quasi-totalité des cas
- Une partie fractionnaire F au format f
- Format n = i + f



Représentation en virgule fixe

- Quelle représentation signée utilisée pour la partie entière ?
  - le complément à 2^n

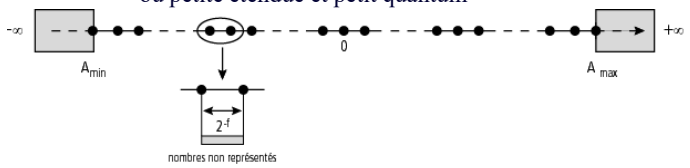


Représentation en virgule fixe

- Exemple
  - n = 4 avec i = 2 et f = 2
    - $1{,}5_{10} = 01{,}10_{2vf}$
- Mais virgule implicite en mémoire
  - pour un gain de place

Représentation en virgule fixe

- La précision
  - appelée aussi quantum $q = B^{-f}$
  - absolue (*i.e.* constante) et non relative
    - ⇒ grande étendue et grand quantum
      ou petite étendue et petit quantum



Représentation en virgule fixe (signée $c2^n$)

- Formule de décomposition

$$A_{10<c2n\_vfx} = \left(a_{i-1} \times -2^{i-1}\right) + \left(\sum_{j=0}^{i-2} \left(a_j \times 2^j\right) + \sum_{k=1}^{f} \left(a_{-k} \times 2^{-k}\right)\right)$$

$$= \left(a_{n-1} \times -2^{n-1}\right) + \left(2^{-f} \times \sum_{j=0}^{n-2} a_{j-f} \times 2^j\right)$$

- Etendue des valeurs

$$-\left(2^{-f} \times 2^{n-1}\right) \le A \le 2^{-f} \times \left(2^{n-1} - 1\right)$$

ou

$$-\left(2^{-f} \times 2^{n-1}\right) \le A \le 2^{i-1} - 2^{-f}$$

Représentation en virgule fixe

- Notion associée
  - sur-dépassement de capacité (*overflow*) positif ou négatif
    - si $|A| > 2^{i-1} - 2^{-f}$
- Attention, pas de sous-dépassement (*i.e.* autour du zéro)

Représentation en virgule fixe

- Avantages/inconvénients
  - \+ calcul simplifié d'où vitesse de calcul élevée
    - utilisation en traitement numérique du signal
  - – quantum q absolu et non relatif = 2^{-f}
    = précision

Méthode de conversion proposée

- Pour la partie entière
  - même méthode que pour les entiers
- Pour la partie fractionnaire
  - multiplication par 2
  - méthode à généraliser pour une base B
    - × B

Exemple de conversion

- Soit à convertir 0,6875 en base 2 (i = 1) :

  $0{,}6875 \times 2 = 1{,}375 \rightarrow 1$ (poids $a_{-1} = 2^{-1}$)
  $0{,}375 \times 2 = 0{,}75 \rightarrow 0$ (poids $a_{-2} = 2^{-2}$)
  $0{,}75 \times 2 = 1{,}5 \rightarrow 1$ (poids $a_{-3} = 2^{-3}$)
  $0{,}5 \times 2 = 1 \rightarrow 1$ (poids $a_{-4} = 2^{-4}$)

D'où le résultat $A = (0{,}1011)_{2vfx}$

Un problème

- Comment faire des calculs mettant en œuvre des nombres avec des ordres de grandeur très élevés ou très petits ?
  - masse de l'électron = $9{,}1 \times 10^{-28}$ g
  - masse du soleil = $1{,}9891 \times 10^{+33}$ g
  - $N_A \approx 6{,}02 \times 10^{23}$ mol^{-1}
- La réponse : la représentation en virgule flottante

Représentation en virgule flottante

- Un nombre peut s'écrire sous la forme :

  $A = M \times B^E$

  en machine, B = 2 d'où $A = M \times 2^E$
- La représentation du nombre A se compose :
  - d'une mantisse M signée au format m
    - elle définit la précision
  - d'un exposant E signé au format e
    - il définit l'ordre de grandeur

  ⇒ format n = m + e

Exemple

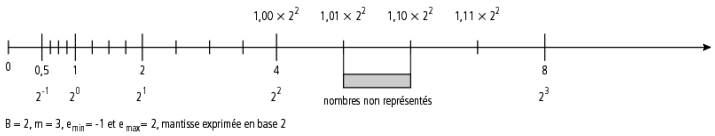
- En base 10, pour faciliter la compréhension :
  $A = 20092010 \times 10^{-4} = 2009{,}2010 \times 10^{-0} =$ etc.
- Un gros problème : plusieurs écritures du nombre !
  - la solution : la normalisation de la mantisse
    en base B : $B^{-1} \leq |M| < 1$
    en base 10 : $10^{-1} \leq |M| < 1$
  - ⇒ mantisse uniquement fractionnaire
  - pour avoir un maximum de chiffres significatifs
  - l'exemple précédent devient :
    $A = 0{,}20092010 \times 10^{4}$

Généralisation de la normalisation de la mantisse

- Facteur de normalisation α
  - $B^{\alpha-1} \leq |M| \leq B^{\alpha}$
  - α = 0 en général :
    $\frac{1}{B} \leq |M| < 1$
  - α = 1 avec la norme IEEE (B = 2) :
    $1 < |M| < 2$
    - pour gagner un bit de précision

Normalisation de la mantisse

- Avantages/inconvénients
  - précision relative améliorée
  - précision absolue réduite
    - ⇒ $A < 0{,}1_2 \times 2^{e(min)}$ non codable
      - zéro en particulier



Représentation en virgule flottante

- Avantages/inconvénients
  - \+ quantum relatif
  - \+ précision adaptable
  - – difficultés d'implémentation
    - bogue de l'instruction FDIV du Pentium Pro (1994)
    - bogue du vol 501 d'Ariane (1996)
  - – valeur nulle non représentée

Représentation en virgule flottante

- Dispositions possibles de l'exposant et de la matrice

mantisse	exposant

ou

exposant	mantisse

- En général,
  - la mantisse est représentée en signe et module (2sm)
  - l'exposant est en complément à 2^n (c2n)

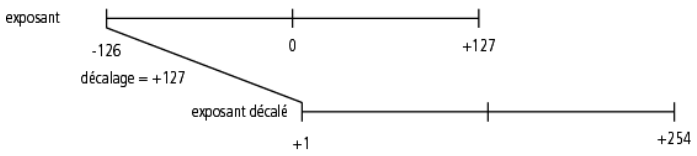
L'exposant décalé (ou biaisé)

- En anglais, *biased exponent* ou E_b
- Objectif : rendre l'exposant E toujours positif
  - comparaison en machine d'exposants plus facile
- Valeur de décalage = *offset*
- Nous avons :

  $E_b = E + \text{offset} = E + (2^{e-1} - 1)$
- Pour e = 8 bits, l'offset est de 127

Exemple

- En simple précision
  - 0 et 255 sont des valeurs spéciales



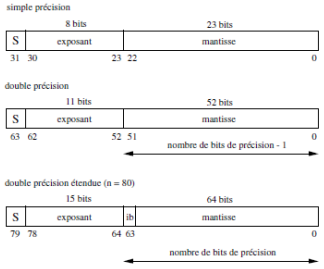
Représentation en virgule flottante

- Formule (avec exposant décalé → $1 < |M| < 2$)

$$|M| = (1, f_{-1} f_{-2} \ldots f_{-(m-2)})_2 = 1 + \sum_{i=0}^{m-2} f_{-i} \times 2^{-i}$$

Les normes IEEE 754 et 854

- Quatre précisions :
  - simple, simple étendue, double et double étendue



Les normes IEEE 754 et 854

- Étendue des valeurs

Format	Etendue normalisée approximative
Simple précision	$1{,}18\ 10^{-38} \leq \lvert A \rvert \leq 3{,}40\ 10^{38}$
Double précision	$2{,}23\ 10^{-308} \leq \lvert A \rvert \leq 1{,}79\ 10^{308}$
Précision étendue	$3{,}37\ 10^{-4932} \leq \lvert A \rvert \leq 1{,}18\ 10^{4932}$

La norme IEEE 754

- Caractéristiques principales

Paramètres	Format			
	simple précision	simple précision étendue	double précision	double précision étendue
Format (n bits)	32	≥ 43	64	≥ 79
Mantisse (p bits)	24	≥ 32	53	≥ 64
Exposant (e bits)	8	≥ 11	11	≥ 15
E_{max}	+127	≥ +1023	+1023	≥ +16383
E_{min}	-126	≤ -1022	-1022	≤ -16382
Décalage exposant	$+127 = (7F)_{16}$	non spécifié	$+1023 = (3FF)_{16}$	non spécifié

La norme IEEE 754

- Implémentée dans le coprocesseur Intel 80x87

Paramètres	Format		
	simple précision	double précision	double précision étendue
Format (n bits)	32	64	80
Mantisse (p bits)	24	53	64
Exposant (e bits)	8	11	15
E_{max}	+127	+1023	+16383
E_{min}	-126	-1022	-16382
Décalage exposant	$+127 = (7F)_{16}$	$+1023 = (3FF)_{16}$	$+16383 = (3FFF)_{16}$

Récapitulatif (1/2)

Format	Précision	Magnitude	Représentation
Entier 16 bits	16 bits	10^4	complément à 2^n
Entier court	32 bits	10^9	complément à 2^n
Entier long	64 bits	10^{18}	complément à 2^n
BCD compacté	18 chiffres	10^{18}	BCD
Réel simple précision	24 bits	$10^{\pm 38}$	virgule flottante
Réel double précision	53 bits	$10^{\pm 308}$	virgule flottante
Précision étendue	64 bits	$10^{\pm 4932}$	virgule flottante

Récapitulatif (2/2)

Format	Format n	Nb chiffres significatifs (décimal)	Etendue normalisée approximative
Entier 16 bits	16	4	$-32768 \leq A \leq +32768$
Entier court	32	9	$-2\ 10^9 \leq A \leq +2\ 10^9$
Entier long	64	18	$-9\ 10^{18} \leq A \leq +9\ 10^{18}$
BCD compacté	80	18	$-99 \cdots 99 \leq A \leq +99 \cdots 99$ (18 chiffres)
Réel simple précision	32	7	$1{,}18\ 10^{-38} \leq \lvert A \rvert \leq 3{,}40\ 10^{38}$
Réel double précision	64	15-16	$2{,}23\ 10^{-308} \leq \lvert A \rvert \leq 1{,}79\ 10^{308}$
Précision étendue	80	19	$3{,}37\ 10^{-4932} \leq \lvert A \rvert \leq 1{,}18\ 10^{4932}$

Conclusion sur la représentation des réels en machine

- ∃ standards
- Limitations
  - portée (de moins en moins vrai)
  - précision
    - risque de valeur approchée
    - erreurs cumulatives de calcul à surveiller
  - implémentation très difficile
    - erreurs potentiellement catastrophiques
  - problème de portabilité lors d'un échange de données