Hugo

Intro : présenta° d'un modèle de rep des entiers, alternative du complément à 2

Plan

I/ Circuits de calcul
1) Modèle
2) Caractéristiques de perf
II/ Représenta° des entiers
1) Complément à B
2) Chiffres signés et redondance
III/ Addition
1) Version naïve
2) Sans redondance
3) Avec redondance, profondeur
IV/ Multiplication
1) Naïve
2) Diviser pour régner
3) Multiplication rapide

I/1) Modèle de circuit

On fixe : - ens de chiffres
- fn°, arith ou logiques, élémentaires : puces (combinées en circuit électronique, en connectant E/S)

Ex : $(a,b) \to (a+b \bmod B, \text{retenue})$
$(u) \to (\neg u)$
$(u,v) \to (u \wedge v)$
$(a,b,c,d) \to (a+b+c+d \bmod B, \text{retenue})$

Circuit : DAG de puces
entrées
sorties

C réalise $(a_1, \dots, a_n) \mapsto (b_1, \dots, b_n)$ lorsque pr chq entrée, l'état stable a les bonnes sorties

I/2) Performance

Critères : taille = # puces
profondeur = lg chemin

Taille → coût construc°
Profondeur → tps pr que circuit se stabilise

Autre critère : énergie consommée. Dépend nb puces (stabilisa° coûte énergie) et profondeur (tps pdt lequel circuit est mobilisé)
→ α Taille + β Taille×Prof

II / Représentation

1) Complément à B

Modèle classique. $a_i \in [\![0; B-1]\!]$. $\sum_{k=0}^{N-1} a_k B^k \subset \mathbb{Z}/B^N\mathbb{Z} \approx [\![\ldots]\!]$

2) Chiffres signés

$a_i \in [\![-\beta; \beta]\!]$, $\beta < B$.
⚠ $\Rightarrow$ redondance potentielle.
Ex : $\overline{2006}_{10} = \overline{1(-8)006}^{10} = \overline{201(-4)}^{10} = \overline{1(-8)01(-4)}^{10}$

[Présentation de représentation Python]

III / Addition

Ça sert à quoi la redondance ?

1) Version naïve de add° $\rightarrow$ ce qu'on apprend en primaire.
On additionne chiffre par chiffre, en propageant la retenue.
$\Rightarrow$ besoin du résultat des chiffres précédent. Profondeur $\Theta(n)$.

2) On peut calculer s'il y aura une retenue $\rightarrow$ Profondeur $\Theta(\log n)$
Pour tt ens de portes, $\exists f : \mathbb{N} \to \mathbb{N}$, $f = \Omega(\log n)$
tq si un circuit C calcule la retenue sortante d'une add° à n chiffres,
profondeur$(C) \geq f(n)$

Lemme : si profondeur$(C) < k$ et $< n$,

[Présenta° code]

[...]