université angers

UTILISATION DE LA LOGIQUE PROPOSITIONNELLE POUR RESOUDRE DES INEQUATIONS LINEAIRES

satō

砂糖

さとう

Mariam Bouzid – Justine Evrard

INTRODUCTION

Compiling finite linear CSP into SAT, 2009

Naoyuki Tamura - Akiko Taga - Satoshi Kitagawa - Mutsunori Banbara

Encodage CSP vers SAT : ***order encoding***

symboles propositionnels ≡ comparaisons de la forme $x \leq a$

≠

sparse encoding

砂糖
さとう

CONTEXTE

CSP : *Constraint Satisfaction Problem*

Limitation aux CSP linéaires sous nombres entiers

⇒ Variables définies sur des sous-ensembles de $\mathbb{Z}$

SAT : *boolean / propositional SATisfiability problem*

$$\text{CSP} \xrightarrow{\textbf{order encoding}} \text{SAT}$$

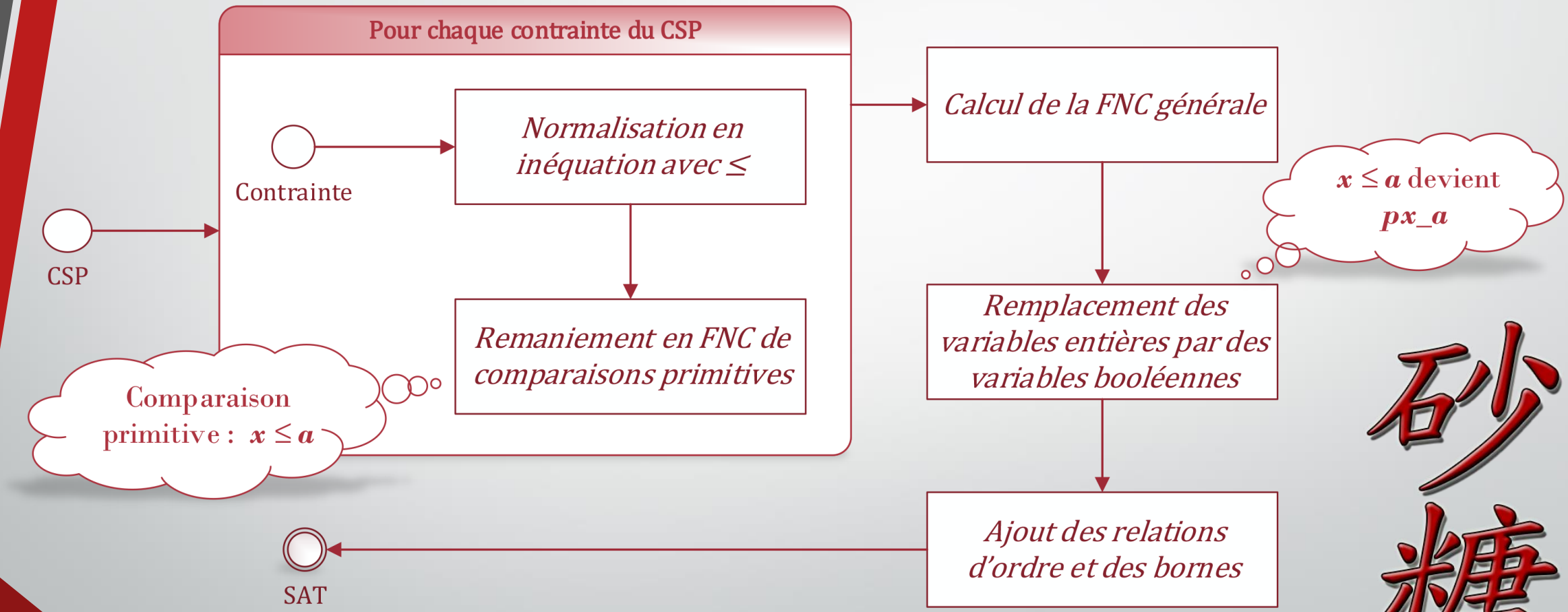
mathématiques → logique propositionnelle

$$\textbf{\textit{Solver SAT}}\textit{ : réponse} = \begin{cases} 'satisfiable' + valuation \\ 'insatisfiable' \end{cases}$$

砂糖
さとう

Justine Evrard & Mariam Bouzid

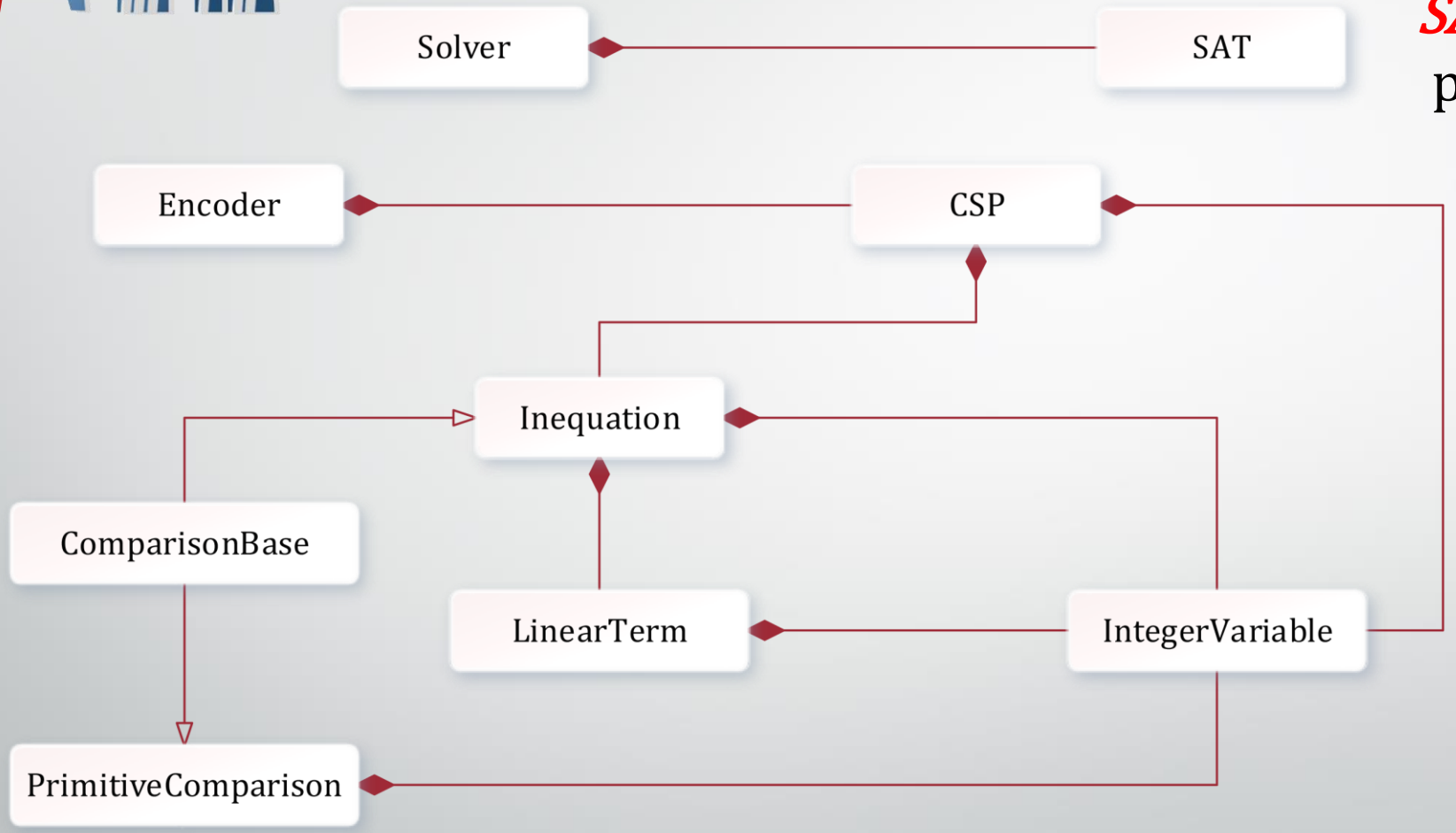
ORDER ENCODING



砂糖
さとう

Justine Evrard & Mariam Bouzid

IMPLEMENTATION



SATŌ = terme japonais pour désigner le sucre

Référence aux projets *sugar, azucar, glucose*...

砂糖
さとう

Justine Evrard & Mariam Bouzid

UTILISATION DU SCRIPT

./sato $x_1\ l_1\ u_1\ \ldots\ x_n\ l_n\ u_n\ ['< contrainte_1 >'\ \ldots]\ \ldots[\ldots' < contrainte_n > ']$

Contrainte : deux expressions linéaires séparées par un des symboles *<=, >=, <, >, !=, =*

```
AVAILABLE OPTIONS:
  -h or --help : display this documentation
  -v : verbose - display solver SAT output
  -i : interprete and display the solver's solutions
  -f <file_name.dimacs> <comment>: only generate the dimacs
file with name 'file_name.dimacs' and a comment 'comment'
```

砂糖
さとう

Justine Evrard & Mariam Bouzid

EXEMPLES D'EXECUTION

Exemple 1 : $y \geq x - 3 \; et \;\; y < 1 - x$

Exemple 2 : $z = x + y + 1$

Satisfiabilité en fonction des domaines des variables…

砂糖
さとう

Justine Evrard & Mariam Bouzid