



---

# RÉSOLUTION DE PROBLÈMES, GÉNÉRATION DE MOTS-CROISÉS

ANDROÏDE M1 – RP

---

Thibault GIGANT  
Laura GREIGE

*Enseignants :*

Patrice PERNY  
Morgan CHOPIN

2015 – 2016

# Table des matières

Introduction	1
1 Modélisation par un CSP et résolution	1
Conclusion	1

## Introduction

### 1 Modélisation par un CSP et résolution

Pour résoudre ce problème, on peut le modéliser comme un problème de satisfaction de contraintes en associant une variable à chaque mot de la grille. Supposons qu'il y ait  $m$  mots dans la grille.

**Variables :**  $x_i$ ,  $\forall i \in \{1, \dots, m\}$

**Domaines :** Soit *dict* un dictionnaire de mots admissibles :

$$D(x_i) = \{X \in \text{dict}\}$$

**Contraintes :** Soit  $l_i$  la taille du mot en  $i$  :

$$\text{len}(x_i) = l_i \tag{1}$$

Pour tous mots  $x_i$  et  $x_j$  qui se croisent à la  $q$ -ième lettre de  $x_i$  et à la  $p$ -ième lettre de  $x_j$ , on a :

$$x_i[q] = x_j[p] \tag{2}$$

Si l'on ajoute la contrainte supplémentaire qu'un même mot ne peut apparaître plus d'une fois dans la grille, il suffit d'ajouter la contrainte *AllDiff* :

$$\text{AllDiff}(x_1, x_2, \dots, x_m) \tag{3}$$

### 2 Expérimentation

### 3 Extension au cas pondéré

## Conclusion