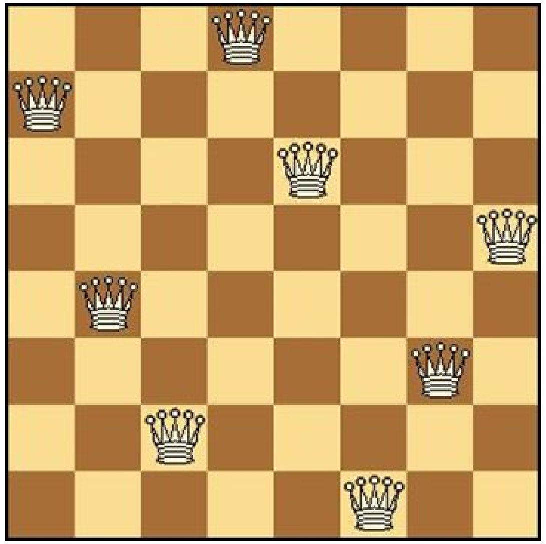
Optimisation et Complexité

« Problème des reines en SAT »

Enseignant : Alain Giorgetti



Avec la Participitation de :

Mehdi Azizi, Elodie Bernard, Mathieu Briland, Anthony Casagrande, Benoit Crivelli, Wentian Huang, Vianney Lotoy Bendenge, Yassin Ousleveh Bileh, Simon Pallais, Cédric Petetin, Alexis Plumet, Pierre Wargnier

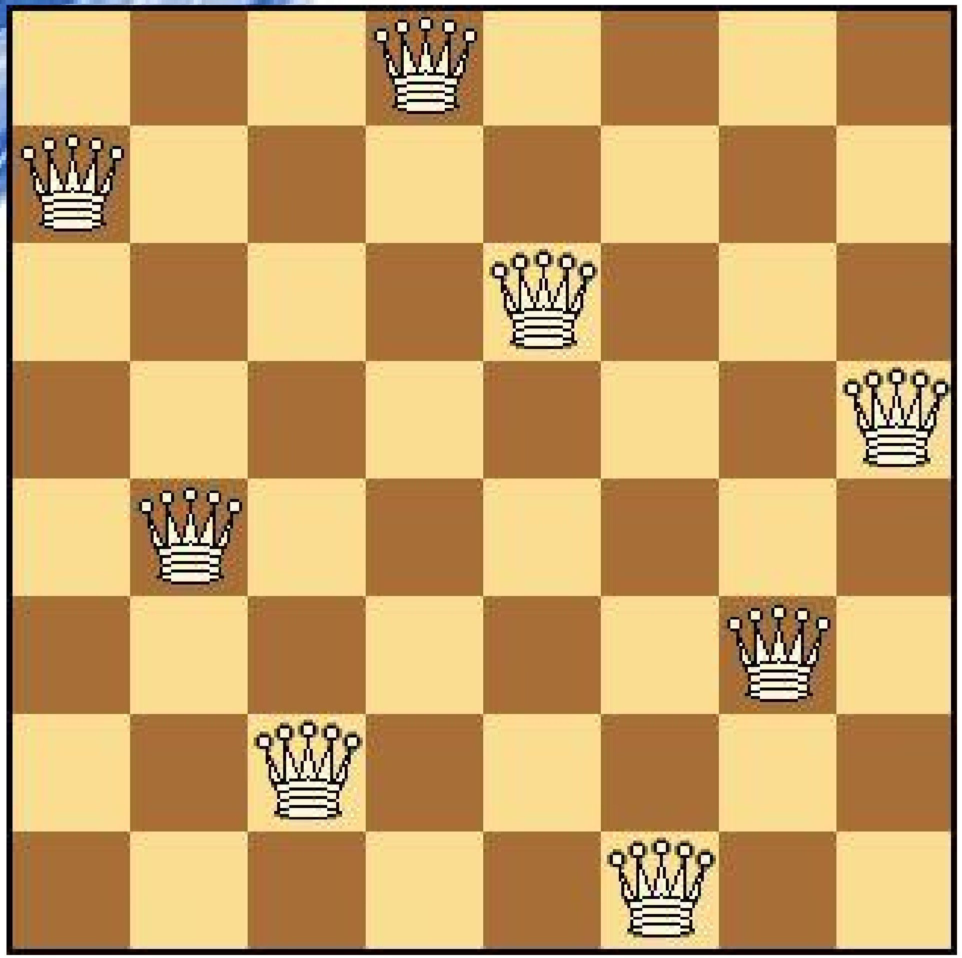
UFC UNIVERSITÉ DE FRANCHE-COMTÉ

Année 2016/2017

Sommaire

1. Formalisation
2. Problème de satisfaisabilité
3. Représentation DIMACS
4. Programmation
5. Expérimentation
6. Solution

1. Formalisation



$\bigvee_{1\leq i,j,k\leq n,\ j<k} (r_{i,j} \land r_{i,k})$

$\bigvee_{1\leq i,j,k\leq n,\ i<k} (r_{i,j} \land r_{k,j})$

$\bigvee_{1\leq i,j\leq n-1,\ 1\leq k\leq n-i,\ 1\leq k\leq n-j} (r_{i,j} \land r_{i+k,j+k})$

$\bigvee_{1\leq i,j\leq n-1,\ 1\leq k\leq n-i,\ 1\leq k\leq j-2} (r_{i,j} \land r_{i+k,j-k})$

$\bigwedge_{1\leq i\leq n} (\bigvee_{1\leq j\leq n} (r_{i,j}))$

2. Problème de satisfaisabilité

$$\Phi = \bigwedge_{1\le i,j,k\le n,\ j\neq k} (\neg r_{i,j} \vee \neg r_{i,k})$$

$$\bigwedge_{1\le i,j,k\le n,\ j\neq k} (\neg r_{i,j} \vee \neg r_{k,j})$$

$$\bigwedge_{1\le i,j\le n-1,\ 1\le k\le n-i,\ 1\le k\le n-j} (\neg r_{i,j} \vee \neg r_{i+k,j+k})$$

$$\bigwedge_{1\le i,j\le n-1,\ 1\le k\le n-i,\ 1\le k\le j-2} (\neg r_{i,j} \vee \neg r_{i+k,j-k})$$

$$\bigwedge_{1\le i\le n} \left(\bigvee_{1\le j\le n} (r_{i,j})\right)$$

3. Représentation DIMACS

Représentation Dimacs dans un fichier .CNF pour n = 4:

p cnf **16 80**
1 2 3 4 **0**
5 6 7 8 0
9 10 11 12 0
13 14 15 16 0
-1 -2 0
-1 -3 0
-1 -4 0 …

16 = Le nombre de variables (les 16 cases de l'échiquier)
80 = le nombre de clauses
0 = représente un ET
‘ ‘ = représente un OU
- = représente une négation

3. Représentation DIMACS

Exemple :

n = 4, contrainte : Au plus une dame par ligne : $\bigwedge_{1\leq i,j,k\leq n,\ j\neq k}(\neg r_{i,j} \vee \neg r_{i,k})$

p cnf 16 80

1 2 3 4 0	**1** $\vee$ **2** $\vee$ **3** $\vee$ **4** $\wedge$
5 6 7 8 0	**5** $\vee$ **6** $\vee$ **7** $\vee$ **8** $\wedge$
9 10 11 12 0	**9** $\vee$ **10** $\vee$ **11** $\vee$ **12** $\wedge$
13 14 15 16 0	**13** $\vee$ **14** $\vee$ **15** $\vee$ **16** $\wedge$

…

4. Programmation

Exemple :

n = 4, Il n'y a pas deux reines sur la même ligne : $\bigwedge_{1\leq i,j,k\leq n,\ j\neq k}(\neg r_{i,j} \vee \neg r_{i,k})$

```
for (i=1; i<=n; i++){
    for(j=1; j<=n; j++){
        k=j+1;
        while(k<=n){
            writer.print("-");
            buf = (i-1)*n + j;
            writer.print(buf +" -");
            buf = (i-1)*n + k;
            writer.print(buf+" ");
            writer.println("0");
            k++;
        }
    }

}
```

→

```
-1 -2 0
-1 -3 0
-1 -4 0
-2 -3 0
-2 -4 0
-3 -4 0
...
...
-14 -15 0
-14 -16 0
-15 -16 0
```

4. Programmation

Exemple : n = 4, Il n'y a pas deux reines sur la même (grande) diagonale descendante:

$$\bigwedge_{1\leq i,j\leq n-1,\ 1\leq k\leq n-i,\ 1\leq k\leq j-2} (\neg r_{i,j} \vee \neg r_{i+k,j-k})$$

```
// Part for the biggest diagonal
for (i=1; i<=n-1; i++){
    for(j=i+1; j<=n; j++){
        writer.print("-");
        buf = i + (i-1)*n;
        writer.print(buf +" -");
        buf = j + (j-1)*n;
        writer.println(buf+" 0");
    }
}
```

-1 -6 0
-1 -11 0
-1 -16 0
-6 -11 0
-6 -16 0
-11 -16 0

4. Programmation

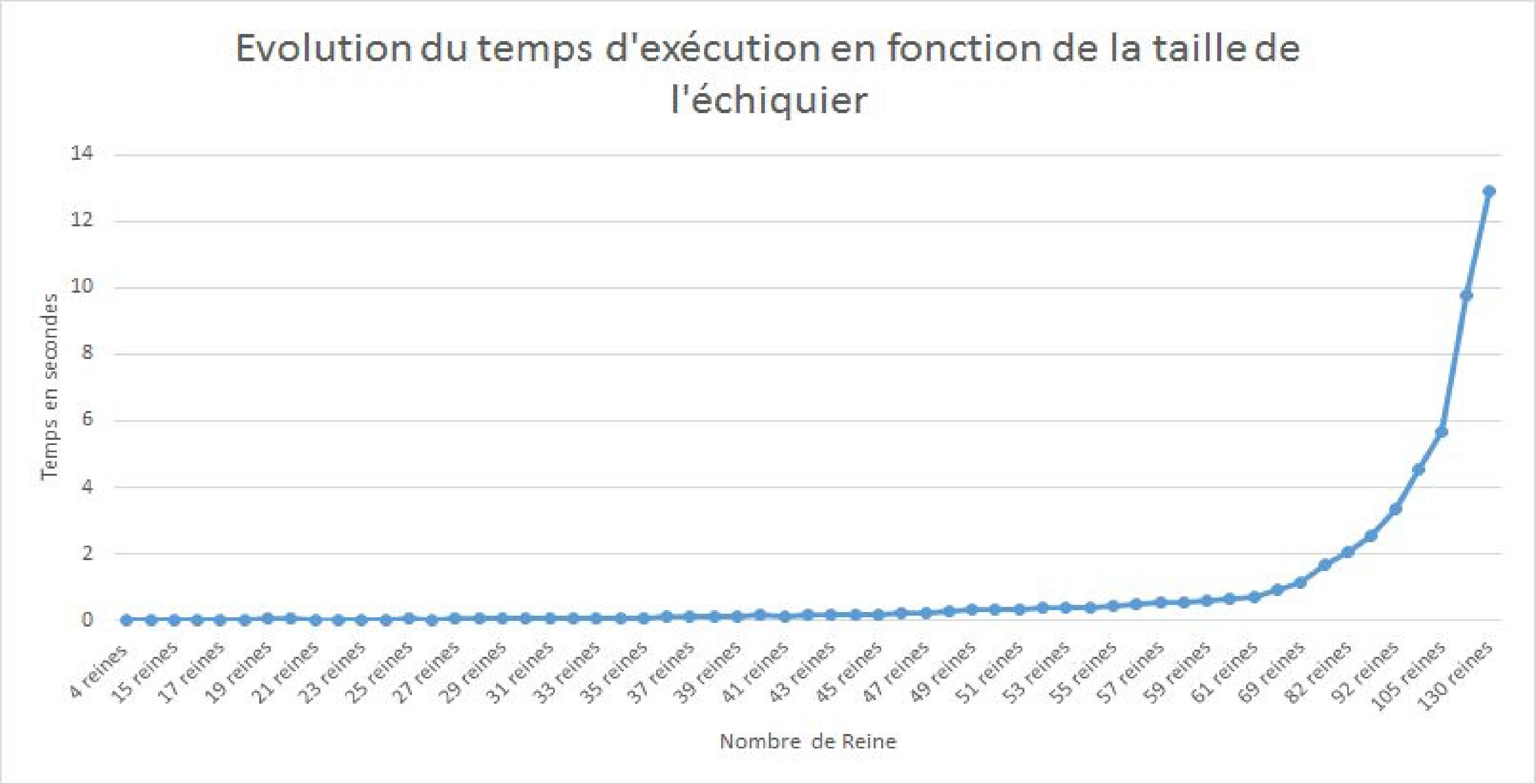
Exemple : n = 4, Il n'y a pas deux reines sur la même (petite) diagonale descendante:

$$\bigwedge_{1\leq i,j\leq n-1,\ 1\leq k\leq n-i,\ 1\leq k\leq j-2} (\neg r_{i,j} \vee \neg r_{i+k,j-k})$$

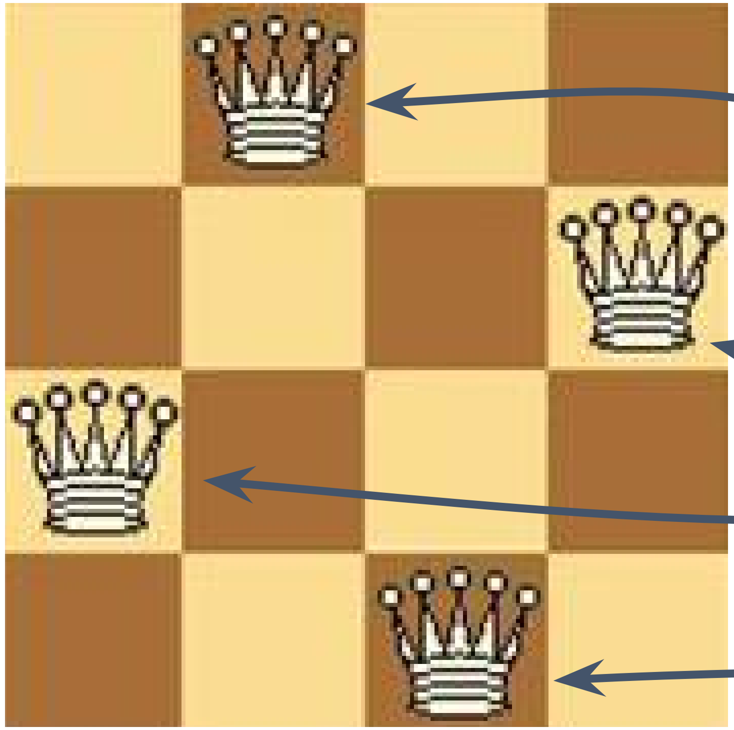
```
// Part for the other diagonals
for(i=2; i<=n-1; i++){
    for(j=i; j<=n-1; j++){
        for(k=j+1; k<=n; k++){
            writer.print("-");
            buf = i + (j-i)*(n+1);
            writer.print(buf +" -");
            buf = i + (k-i)*(n+1);
            writer.println(buf+" 0");
        }
    }
    for(j=i; j<=n-1; j++){
        for(k=j+1; k<=n; k++){
            writer.print("-");
            buf = i + (j-i)*(n+1) + (i-1)*(n-1);
            writer.print(buf +" -");
            buf = i + (k-i)*(n+1) + (i-1)*(n-1);
            writer.println(buf+" 0");
        }
    }
}
```

-2 -7 0	-5 -10 0
-2 -12 0	-5 -15 0
-7 -12 0	-10 -15 0
-3 -8 0	-9 -14 0

5. Expérimentation



6. Solution



Solution trouvée pour n=4

-1 2 -3 -4 -5 -6 -7 8 9 -10 -11 -12 -13 -14 15 -16 0