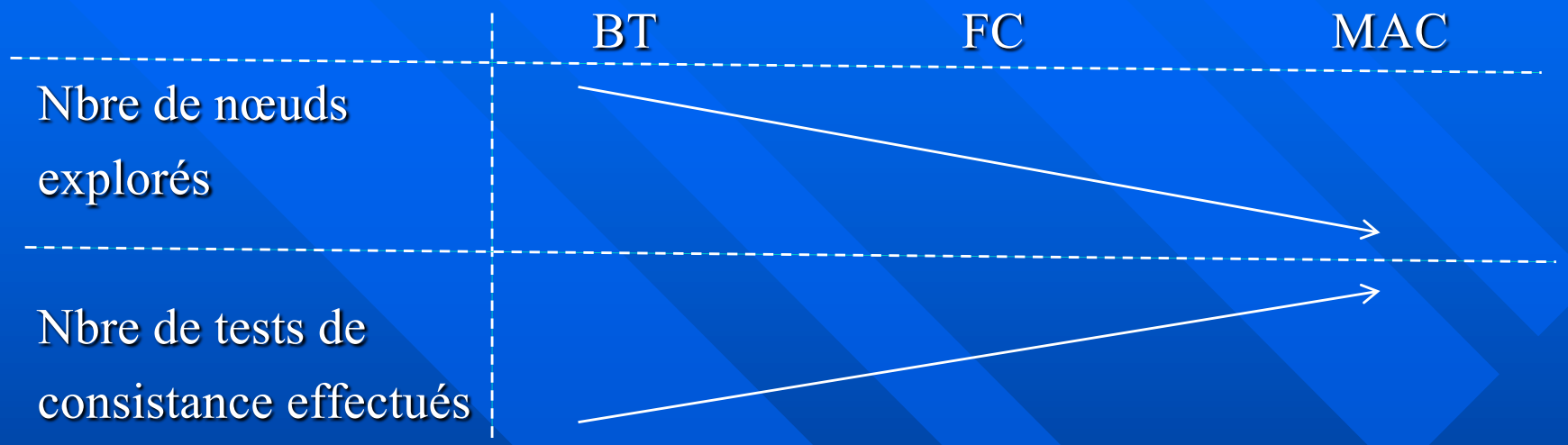
Partie 2 – Problèmes de satisfaction de contraintes

2.2 LA RESOLUTION

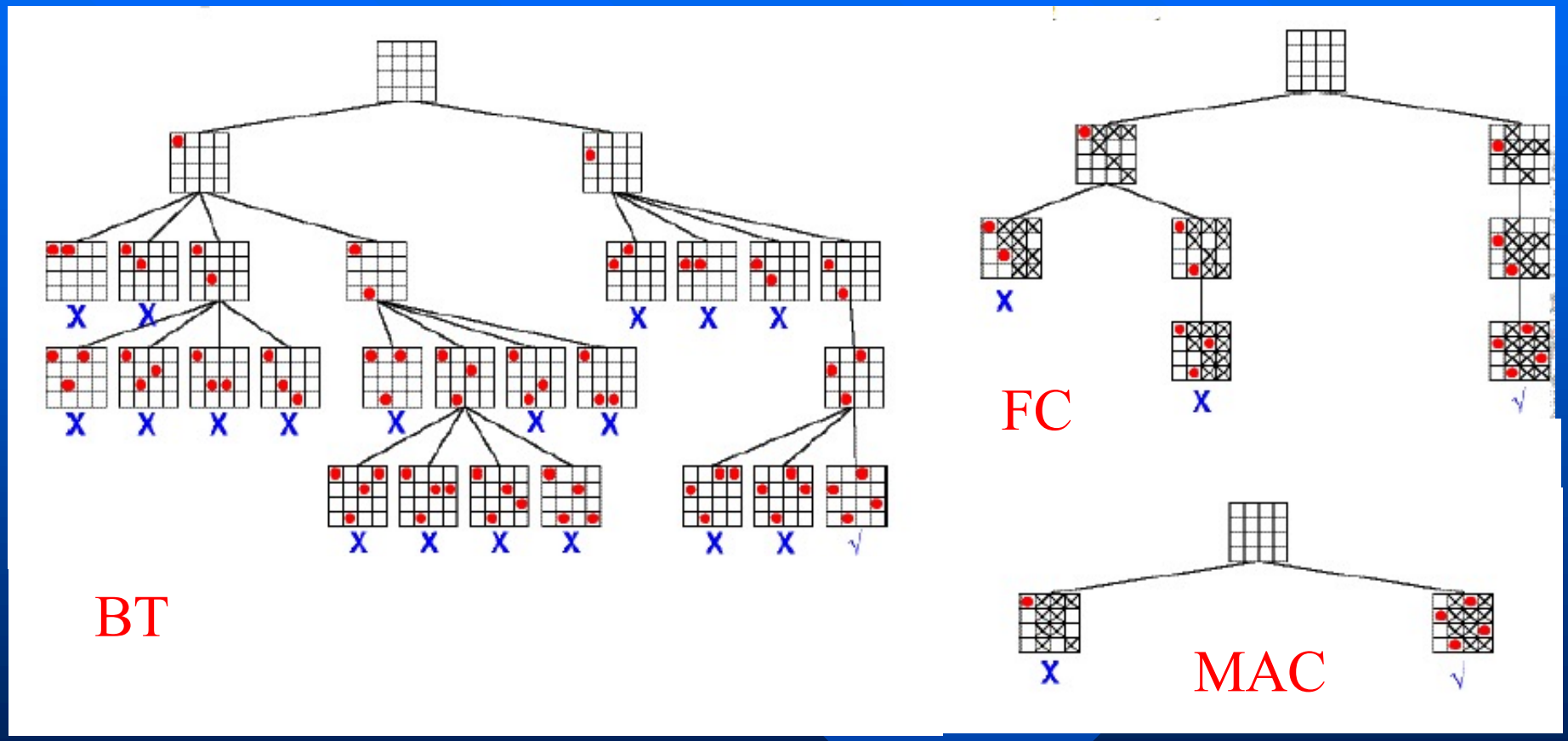
2.2.4 EXPÉRIMENTATION

Efficacité des algorithmes



➔ Trouver un compromis entre filtrage et recherche

Exemple sur les 4-reines



Comparaison des méthodes

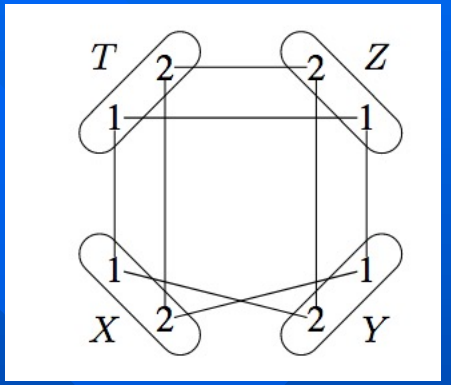
- La résolution de CSP se heurte à une explosion combinatoire qui peut engendrer des temps de résolution très long
- Le problème d'existence d'une solution à un CSP est NP-complet
  - La théorie de la complexité estime le nombre d'instructions à exécuter pour résoudre la (pire) instance en fonction de sa taille
  - On sait donc que l'on ne peut pas résoudre en temps polynomial toutes les instances du problème
- Mais toutes les méthodes et heuristiques de choix ne se valent pas !
- Nécessité de benchmarks pour comparer les méthodes
  - Soit un jeu de CSP représentatif d'un type de pb réel
  - Soit un échantillon aléatoire de CSP

Tirage aléatoire de réseaux

- Les réseaux ne sont pas d'égales difficultés
  - Un réseau peu contraint aura beaucoup de solutions => on détectera rapidement une solution
  - A l'opposé très contraint n'aura pas de solution => on détectera rapidement qu'il n'a pas de solution
  - Entre les deux, il existe des réseaux difficiles pour lesquels on a autant de chance d'avoir une solution que de ne pas en avoir et pour lesquels ce sera donc a priori plus long de trouver une solution (ou de vérifier qu'il n'y en a pas)
- On cherche à comparer les méthodes sur les instances difficiles pour observer de « vraies » différences de comportement
  - Pour cela, on fixe certains paramètres de génération aléatoire et on en fait évoluer un pour chercher la « zone » où on passe de réseaux à bcp de solutions à des réseaux avec pas de solutions.

Paramètres des réseaux

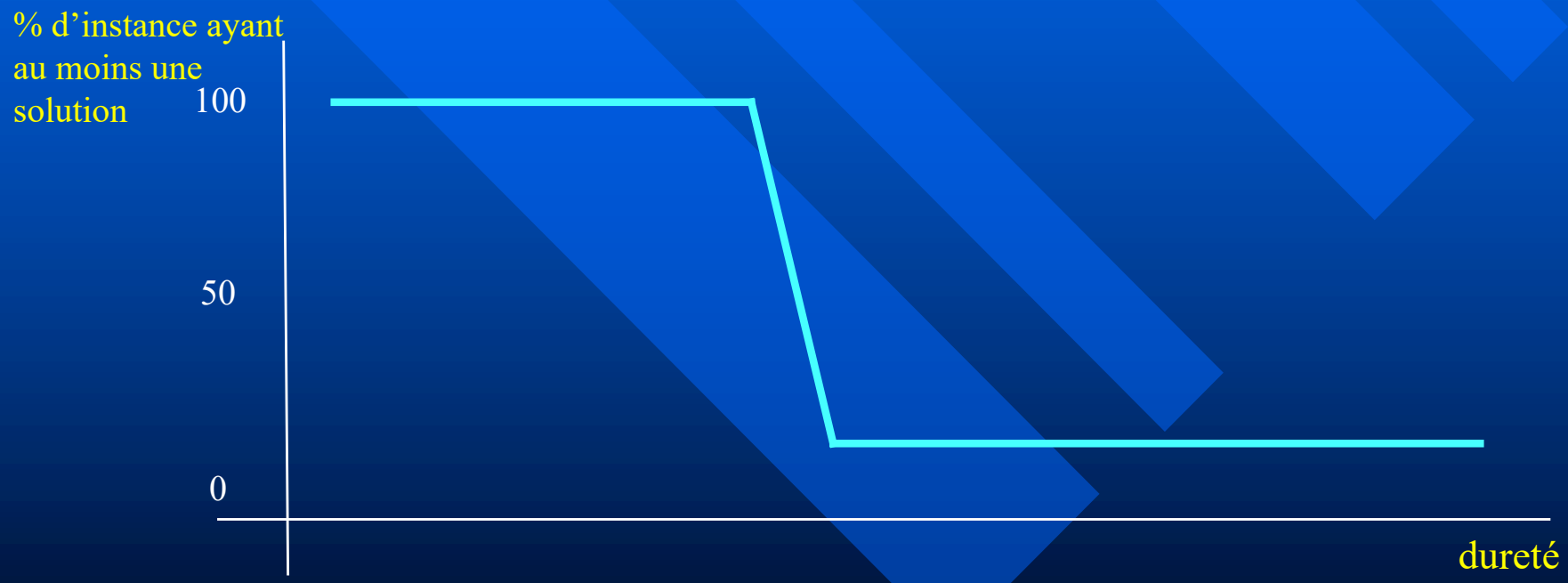
- Une classe d'instance est décrite par 5 paramètres < k, n, d, e, t >
  - k : l'arité (maximale) des contraintes
  - n : le nombre de variables
  - d : la taille des (du plus grand) domaines
  - e : le nombre de contraintes
    - » D : la densité e/C^k_n
  - t : la dureté des contraintes, définie comme le rapport I/T où
    - » I est le nombre de tuples interdits pour la contrainte (produit des tailles des domaines – nombre de tuples de la contraintes)
    - » T est le nombre maximal de tuples possibles pour la contrainte (produit des tailles des domaines)



- Généralement, on fixe les 4 premiers et on fait varier la dureté qui doit directement jouer sur le nbre de solutions
  - Il faut cependant faire des expés pour différentes densités de réseau

Caractérisation des instances difficiles

- On tire pour chaque valeur de dureté nb réseaux et on cherche combien ont une solution, soit s leur nombre => s/nb définit la difficulté du réseau
- On trace une courbe dureté/difficulté.



- On observe généralement un changement brutal appelé la transition de phase.

Comparaison expérimentale

- On mesure en temps CPU (dépendant de la machine)
  - Temps moyen passé pour un test de contrainte
    - » Important puisque ce test est souvent réalisé
  - Temps moyen de résolution des instances d'un niveau de dureté
- On mesure en nombre
  - de nœuds développés
  - de tests de contraintes faits
- Généralement les instances difficiles sont celles sur lesquelles les temps de calcul augmentent et sont donc plus à même de montrer des vrais différences de comportement entre méthodes/heuristiques

Expérimentation <2,35,17,249,t>