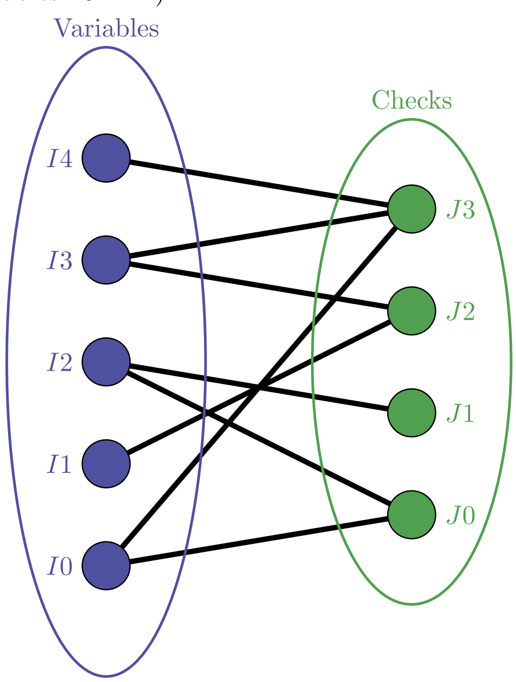
Telecom 202b, Novembre 2021
Enseignant: Aslan Tchamkerten

1 Expander codes (Sipser & Spilman, 1994)

Tous code linéaire C de matrice de parité H peut se représenter comme un graph biparti avec 2 ensembles de noeuds, à gauche les noeuds “variable”, à droite les noeuds ”parités” avec un noeud i à gauche est connecté à un noeud à droite j si $H_{ij} = 1$. Un vecteur x appartient au code C ssi

$$\sum_{i:x_i \neq 0} H_{ij} = 0$$

pour tout noeud j. Par exemple, le code ci-dessous contient le mot code 01011 (assignés aux variables $I0 - I4$).



Definition 1 (Code expander) *Un code expander $(n, m, D, \gamma, \alpha)$ est un code dont la graphe biparti de la matrice de parité satisfait:*

- *le nombre de variables est n,*
- *le nombre de checks est m (donc le taux est $1 - m/n$),*
- *le graphe est D-régulier à gauche, i.e., chaque variable est connectée à exactement D checks,*

- *pour tous sous ensemble S de variable avec $|S| \leq \gamma n$ on a $N(S) \geq \alpha|S|$, où $N(S)$ représente l'ensemble des voisins de S (i.e., l'ensemble des checks qui sont connectés à au moins une variable de S).*

On note qu'on a clairement $\alpha \leq D$. Le théorème suivant indique que cette borne peut être approchée arbitrairement.

Théorème 1 *$\forall \varepsilon > 0$, $m \leq n$, il existe $\gamma > 0$ et $D \geq 1$ t.q. il existe un code expander $(n, m, D, \gamma, D(1 - \varepsilon))$.*

Preuve 1 *Voir TD5 pour une preuve avec la méthode probabiliste.*

Les codes expander sont intéressant en codage car γ est lié à la distance minimale du code $\Delta(C)$:

Théorème 2 *Soit $(n, m, D, \gamma, D(1 - \varepsilon))$ un code expander avec $\varepsilon < 1/2$. Alors*

$$\Delta(C) \geq 2\gamma(1 - \varepsilon)n.$$

Preuve 2 *Voir TD5.*

Corollaire 1 *Pour taux $R < 1$ il existe un code expander de distance minimale $\Theta(n)$.*

Les codes expander représentent la première construction autre que la concaténation qui permette une distance minimale $\Theta(n)$.

Décodage (bit-flipping)

Soit $y_1, y_2, \ldots, y_n$ le mot code reçu qui correspond aux valeurs des variables et soit $VS(i)$ et$VNS(i)$ l'ensemble des checks voisins de la variable i qui sont satisfait et non satisfait, respectivement. Initialiser $z_i = y_i$ pour tout i.

Algorithme: While there exists variable i such that $|VS(i)| < |VNS(i)|$ flipper z_i. If not possible output z.

En clair, si une variable pose plus de problèmes qu'elle en résout, sa valeur est changée.

Théorème 3 *Soit $0\varepsilon < 1/4$ et soit $(n, m, D, \gamma, D(1 - \varepsilon))$ un code expander. Alors l'algo de bit-flipping corrige tout pattern d'au plus $\gamma(1 - 2\varepsilon)n$ erreurs en temps $O(n^2)$.*

Preuve 3 *Voir TD5.*